Zygote Scoring,
a new approach

Stud: Antoine Beuchat
Prof: Michael Unser
Assist: Carlos Oscar Sánchez Sorzano

Biomedical Imaging Group (EPFL-STI-IOA-LIB)
Centro Nacional de Biotecnología (UAM-CSIC-CNB)

MASTER'S DIPLOMA PROJECT 2004-2005

Multivariate statistical analysis and classification of zygotes

Student:	Antoine Beuchat
Director:	Prof. Michael Unser, Biomedical Imaging Group, Ecole Polytechnique Fédérale de Lausanne (EPFL)
Assistant:	Dr. Carlos Oscar Sánchez Sorzano, Centro Nacional de Biotecnología, Consejo Superior de Investigaciones Científicas (CSIC)
Collaborators:	Dr. Alfred Senn, Dr. Françoise Urner, Laboratoire d'Andrologie et de Biologie de la Reproduction, Unité de Médecine de la Reproduction (CHUV)

(Antoine Beuchat)

Contents

1 Introduction **4**

2 Data Collection **7**
2.1 Data . . . 7
2.1.1 PNS dataset . . . 8
2.1.2 IJPM dataset . . . 8

3 Data examination **13**
3.1 Normality . . . 13
3.1.1 Shapiro-Wilk test for normality . . . 14
3.1.2 Multivariate normality . . . 14
3.2 Quality of the data and outliers . . . 15
3.2.1 "Cleaning" data stage . . . 15
3.2.2 Visual detection . . . 15
3.2.3 Automated outlier detection . . . 16
3.2.4 Automated detection comparison . . . 17

4 Individual Classification Power **19**
4.1 Categorical data . . . 19
4.1.1 Chi-Square test for independence . . . 19
4.1.2 Summary . . . 21
4.2 Continuous data - Measure of association and correlation . . . 21
4.2.1 Pearson product-moment correlation . . . 22
4.2.2 Kendall's tau correlation . . . 23
4.2.3 Kolmogorov-Smirnov goodness-of-fit test . . . 25
4.2.4 Correlation Summary . . . 26
4.2.5 PCA . . . 27

5 Supervised Classifiers **32**
5.1 Classification accuracy . . . 32
5.2 Comparison between classification strategies . . . 35
5.3 LDA . . . 35
5.3.1 Mathematical background . . . 35
5.3.2 Results . . . 37
5.3.3 Conclusion . . . 40
5.4 Trees . . . 42
5.4.1 Mathematical background . . . 42
5.4.2 Results . . . 45

5.4.3 Conclusion 46
5.4.4 Mathematical background 49
5.5 SVM 49
5.5.1 Mathematical background 49
5.5.2 Results 50
5.5.3 Conclusion 52
5.6 Random Forest 53
5.6.1 Mathematical background 54
5.6.2 Results 54
5.6.3 Conclusion 55
5.6.4 Variable importance 57
5.6.5 Supervised Classification with conservation type 59

6 Improvement 62

7 Conclusion 67

A Appendix A 75

B Appendix B 76

C Appendix C 77

Chapter 1

Introduction

The Swiss law (*LPMA*)[1] on medically assisted procreation does not allow to develop more than three embryos in vitro beyond the pronuclear stage, but it permits cryopreservation of supernumerary fertilised oocytes prior to pronuclear membrane breakdown. Identification of potentially viable embryos is thus limited either to the oocytes prior to fertilisation or to the pronuclear stage zygote . The present study was focused on the pronuclear stage zygotes (see Figure 1.1). The aim was initiated to assess if morphological characteristics of zygotes could be used as markers of future embryo developmental competence using statistical tools. Thus, allowing us to transfer only the embryos that would result in the highest pregnancy rates while minimizing the high order multiple pregnancies.

Figure 1.1: Pronuclear stage zygotes, one pronucleus contains the genetic material from the sperm and the other pronucleus contains the genetic material from the oocyte

Human embryos produced in vitro have long been recognised to present few characteristics able to predict their implantation ability. The most widely adopted strategy to choose viable embryos is to rely on the number of blastomeres and the grade of the embryos at the time of embryo transfer [1, 2, 3]. However, these morphological aspects do not correlate sufficiently with the embryonic viability to allow an univocal microscopic recognition of the embryos able to produce a viable pregnancy. A number of strategies have thus been proposed to improve the prognostic evaluation of

[1] Loi Fédérale sur la Procréation Médicalement Assistée, 18th Decembre, 1998

embryo viability, including selection of early cleaving embryos [2, 4, 5], culture up to the blastocyst stage [6] and scoring of pronucleate stage zygotes [7, 8, 9, 10, 11]. No study has so far specifically attempted to evaluate the contribution of morphological characteristics automatically detected by an advanced image analysis tool.

In early 2003, a Plug-in [12] using *JAVA*^{TM} was developed by Antoine Beuchat for *ImageJ* [2] software in the biomedical imaging group (BIG) [3] under the supervision of Dr. Sorzano. Images captured under Hofman modulation of the *LABR* laboratorie were used. Information resources, including systems, infrastructure and data, were shared while respecting confidentiality. The Plug-in aims at detecting on digital images contrast morphological characteristics automatically and exporting a database file for statistical analysis.

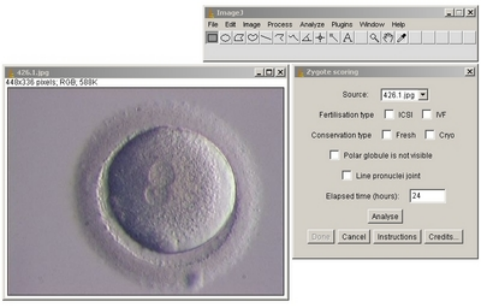


Figure 1.2: ImageJ Plug-In for automated morphological measurements

We compare the performance of several classes of statistical methods for the classification of zygotes at the two pronuclei stage based on digital images. These methods are linear discriminant analysis, classification trees, support vector machine, and random forest. Methods are applied to samples from the *LABR* [4] clinic at Lausanne's University Hospital (CHUV).

Our aim was to analyse the patterns of the zygotes at the two pronuclei stage and to compare the prognostic value of several morphometric parameters of fresh and frozen-thawed zygotes in terms of pregnancy rates. This analysis should help selecting out efficiently the zygotes with higher implantation ability and, thus, reducing the number of transfers necessary to achieve pregnancy and the number of multiple gestations.

The report is divided as follows. In Chapter 2, we introduce the data collection and the main notation. In Chapter 3, we examine the data and asses its quality to get a overview of its structure. Chapter 4 we study the power of each individual feature to classify. Chapter 5 brings the proposal of supervised classifiers including the more commonly used with assessment of the results. Chapter 6 apply the best classifier and we compute the probability of having a pregnancy with realistic values of the *LABR* to get closer to the real situation. Finally, Chapter 7 discusses the better model strategy and concludes. An appendix provides some plots of the data. This analysis should demonstrate if the features are related to the implantation potential of the embryo and,

[2] ImageJ is being developed at the National Institutes of Health of the United States, this software is not subject to copyright protection and is in the public domain

[3] Laboratory in The Institute of Applied Optics (IOA) of the Swiss Federal Institute of Technology (EPFL - Lausanne)

[4] Reproductive Medicine Unit, Department of Gynaecology and Obstetrics, CHUV, Lausanne, Switzerland

thus, confirm (or not) that the information about implantation is contained in the image of the pronuclear zygotes. Data show that at the pronuclear zygote levels a series of criteria can help selecting out embryos with significantly higher implantation abilities.

Chapter 2

Data Collection

2.1 Data

All fresh embryo transfers were performed on day 2 without any embryo selection. All supernumerary zygotes were frozen before syngamy using a slow freezing protocol [13] and were thawed 24-28 h prior to transfer, 3 days after the LH peak in natural cycles. The implantation potential were used as the main end points and submitted to statistical evaluation.

As we have stated above, our aim in this study is to make morphometric measurements and observations based on digital images (640x480 with approximately 60 Ko and 24bit) of oocytes in order to predict the likely outcome. To achieve this two methods are available, one of them is to use a plug-in for *ImageJ* developed at the *EPFL* in collaboration with the *LABR* Laboratories. The collected dataset will be labeled as **IJPM** for **I**mageJ **P**ronuclear Morphometric **M**easurements. The other method is a visual inspection of six zygotes characteristics labeled **PNS** for **P**ronuclear **S**core, each characteristic can take values from 1 to 3 (from worst to best). Tables 2.1 and 2.2 are an overview of the PNS and IJPM dataset, respectively.

Both methods (IJPM and PNS) are based on the same data collected from May 2001 to December 2002. A total of 98 fresh and 110 frozen-thawed zygotes observed under Hofman contrast (300x) were photographed at the two pronuclei (2PN) stage. Zygotes leading to pregnancy and not leading to pregnancy are labeled as "good" and "bad", respectively. Within these 208 zygotes a total of 124 are "bad" and 84 are "good". As two zygotes are transfered on day 2, only 84 zygotes leading to multiple gestations were considered, thus we are sure that both of inseminated zygotes lead to pregnancy. Difference between fresh zygotes and frozen-thawed is so large (see Section 4.1.2), that in our classification studies we will also analyze zygotes separately depending on whether they are "fresh" or "cryo"-preserved.

2.1.1 PNS dataset

The six zygotes characteristics labeled as **PNS** are

1. centering of the two pronuclei (*posPron*)
2. proximity of the pronuclei (*proxPron*)
3. number of nucleoli (*nbNucl*)
4. polarization and realignment of the nucleoli (*distrNucl*)
5. cytoplasmic halo (*cortReaction*)
6. orientation of the pronuclei in respect to the polar bodies (*gpAlign*)

The CPNS (sum over the six characteristics) may thus vary from 6 to 18.

Image	Icsi	posPron	proxPron	nbNucl	...	Cryo	Class
01.1041-1.bmp	yes	3	3	2	...	no	good
02.1047-1.bmp	yes	3	2	1	...	yes	bad
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2.1: Extract of PNS dataset (208 x 9), with three categorical features

2.1.2 IJPM dataset

Image	Icsi	posOfPronSum	proxOfPronCent	...	Cryo	Class
01.1041-1.bmp	yes	34.96	69.85	...	no	good
02.1047-1.bmp	yes	24.1	47.2	...	yes	bad
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 2.2: Extract of IJPM dataset (208 x 59)

Features from IJPM dataset are extracted with the plug-in, Figure 2.1 and 2.2 illustrates the detection process. Figure 2.1 image is helpful throughout this report if you are unfamiliar with biological terminology. The choice of which variables to measure is crucial to successful pattern recognition [14]. Variables selection are based on the experience of the *LABR* staff. We also try to get close to the well-known PNS features. Tables **??**,**??** and 2.5 summarize our features.

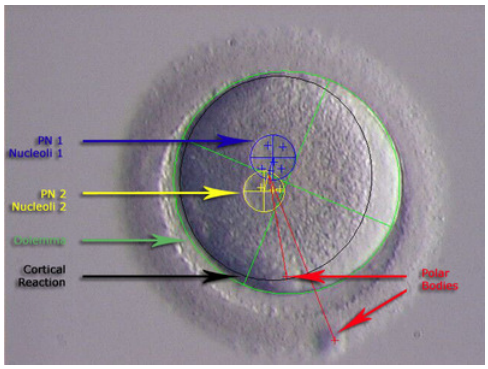


Figure 2.1: Pattern recognition made with *ImageJ* Plug-In

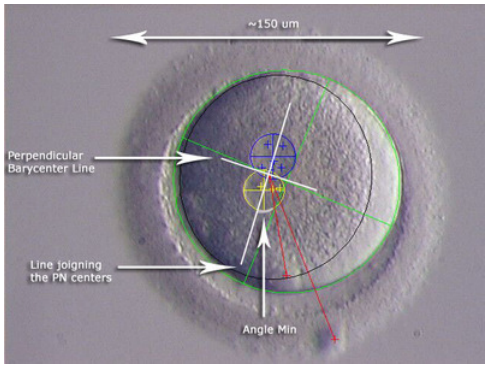


Figure 2.2: Useful labels for features extraction

	Feature	description
1	Title	Name of the digital image
2	Icsi	Fertilisation type in vitro (=no), or Intra-Cytoplasmic Sperm Injection ICSI (=yes) (user input)
3	polarGlobal	detect presence of Polar Bodies (user input)
4	posOfPronSum	mean of distances between the center of the Oolemma and the barycenter of pronuclei
5	posOfPronRel	posOfPronSum divided by the Oolemma radius
6	proxOfPronCent	distance between two centers of the pronuclei
7	proxOfPronRel	proxOfPronCent divided by the sum of the radius of pronuclei (if < 1 pronuclei they overlap)
8	nucleoles1	number of nucleoli in pronucleus #1 considering nucleoles1 always greater than nucleoles2
9	nucleoles2	number of nucleoli in pronucleus #2
10	nucleolesTiny	same as nucleoles1 considering radiusPronBig always greater than radiusPronTiny
11	nucleolesBig	same as nucleoles1 considering radiusPronBig always greater than radiusPronTiny
12	sumOfNucl	sum of nucleoli1 and nucleoli2
13	diffOfNucl	difference of nucleoli1 and nucleoli2
14	dispOfNucl1Avg	mean of the distance between nucleoli in pronucleus #1 and their gravity center
15	dispOfNucl2Avg	mean of the distance between nucleoli in pronucleus #2 and their gravity center
16	dispOfNuclTinyAvg	same as dispOfNucl1Avg considering radiusPronBig always greater than radiusPronTiny
17	dispOfNuclBigAvg	same as dispOfNucl1Avg considering radiusPronBig always greater than radiusPronTiny
18	distOfNucl1ToPronLine	sum of the distances between nucleoli in pronucleus #1 and the perpendicular barycenter line
19	distOfNucl2ToPronLine	sum of the distances between nucleoli in pronucleus #2 and the perpendicular barycenter line
20	distOfNuclTinyToPronLine	same as distOfNucl1ToPronLine considering radiusPronBig always greater than radiusPronTiny
21	distOfNuclBigToPronLine	same as distOfNucl1ToPronLine considering radiusPronBig always greater than radiusPronTiny
22	distOfNucl1ToLineAvgRel	distOfNucl1ToPronLine divided by the number of nucleoli in pronucleus #1
23	distOfNucl2ToLineAvgRel	distOfNucl2ToPronLine divided by the number of nucleoli in pronucleus #2
24	distOfNuclTinyToLineAvgRel	same as distOfNucl1ToLineAvgRel
25	distOfNuclBigToLineAvgRel	same as distOfNucl1ToLineAvgRel considering radiusPronBig always greater than radiusPronTiny
26	lineVarOfNucl1Avg	mean of the distance between the regression line of nucleoli in pronucleus #1 parallel to the perpendicular barycenter line (NA if only one nucleoli in pronucleus #1)

Table 2.3: List of IJPM features

	Feature	description
27	lineVarOfNucl2Avg	mean of the distance between the regression line of nucleoli in pronucleus #2 parallel to the perpendicular barycenter line NA if only one nucleoli in pronucleus #1)
28	lineVarOfNuclTinyAvg	same as lineVarOfNucl1Avg considering radiusPronBig always greater than radiusPronTiny
29	lineVarOfNuclBigAvg	same as lineVarOfNucl1Avg considering radiusPronBig always greater than radiusPronTiny
30	corticalReaction	ratio of the Oolemma to the Cortical Reaction (cytoplasmic halo) radius
31	alphaMin	minimum angle between two intercepting lines formed by the pronucleus centers and the polar bodies #1 to the barycenter of pronuclei line (always < 90°), considering polar bodie #1 having an inferior angle to polar bodie #2
32	alphaMax	maximum angle between two intercepting lines formed by the pronucleus centers and the polar bodies #1 to the barycenter of pronuclei line (always < 90°)
33	alphaBetween	alphaMax-alphaMin
34	alpha3D1	same angle as alphaMin considering the polar bodie #1 being in a sphere
35	alpha3D2	same angle as alphaMin considering the polar bodie #2 being in a sphere
36	OolemmaA	axe A of the Oolemma ellipse, considering OolemmaA always greater than OolemmaB
37	OolemmaB	axe B of the Oolemma ellipse
38	OolemmaRatio	ratio of the OolemmaA to the OolemmaB
39	CorticalA	axe A of the Cortical Reaction ellipse, considering CorticalA always greater than CorticalB
40	CorticalB	axe B of the Cortical Reaction ellipse
41	CorticalRatio	ratio of the CorticalA to CorticalB
42	Ellipse.angle	angle between OolemmaA and OolemmaB
43	ellDistCentersRel	ratio of distance between centers of Oolemma and Cortical Reaction to Oolemma radius
44	ellDistBetweenRel	ratio of largest distance from center of Oolemma to the Cortical Reaction to the Oolemma radius
45	radiusPron1	radius of pronucleus #1
46	radiusPron2	radius of pronucleus #2
47	radiusPronTiny	radius of pronucleus #1 radiusPronBig always greater than radiusPronTiny
48	radiusPronBig	radius of pronculeus #2
49	radiusPron1Rel	ratio of radiusPron1 to the Oolemma radius
50	radiusPron2Rel	ratio of radiusPron1 to the Oolemma radius
51	radiusPronTinyRel	ratio of radiusPronTiny to the Oolemma radius
52	radiusPronBigRel	ratio of radiusPronBig to the Oolemma radius

Table 2.4: list of IJPM features (continued)

	Feature	description
53	ZS1.Centering	centering of the two pronuclei
54	ZS2.Proximity	proximity of the pronuclei
55	ZS.3.Nb.of.nucl	number of nucleoli
56	ZS4.Line.dispersion	polarization and realignment of the nucleoli
57	ZS.4bis.Line.position	polarization and realignment of the nucleoli
58	ZS6.GP.avg.if.90	orientation of the pronuclei in respect to the polar bodies
59	Cryo	determines if the zygote was cryo-preserved (=yes) or not (=no) (user input)
60	**class**	determines if the zygotes leads to pregnancy (=good) or not (=bad) (user input)

Table 2.5: list of IJPM features (continued)

Chapter 3

Data examination

The initial examination of the data is one of the most important parts of the data analysis cycle [14], it constitutes the first phase of the analysis and comprises three parts:

1. checking the quality of the data and "cleaning" data stage;
2. univariate and multivariate analysis (for example producing plots of the data in order to get a feel for their structure)
3. calculating summary statistics.

There are several factors that degrade data quality, the main ones being due to errors, outliers and missing observations. Errors may occur in several ways. They may be due to malfunctions in recording equipment, for example transcription errors. Some errors may be difficult to detect, particularly if the value in error is consistent with other observations. Alternatively, if the error gives rise to an outlier (an observation that appears to be inconsistent with the remainder of the data) then a simple range test on each variable may pick it up. Missing values can arise in a number of different ways and it is important to know how and why they occur. Extreme care must be taken in the coding of missing values, not treating them as special numerical values if possible.

Normality tests and outlier detection are intimately intermingled: normality tests should not include outliers since the results might be severely influenced; on the other hand, automated outlier detection relies on the knowledge of whether a feature is normally distributed or not. Therefore, these two steps are iteratively intermixed until a reliable estimate of the normality of a feature is achieved and the corresponding outliers detected. In the discourse of this report, we discuss first the normality test, and then we study the outlier detection.

3.1 Normality

Our data can be separated in two classes, the first class is the **"good zygotes"** and the second one is the **"bad zygotes"**. The distribution of our features has the contribution from the two classes. Thus, the non normality of a feature may be seen as a good sign. Non normality will be preferable to make an accurate individual classification.

The conventional test for normality when the mean $\overline{x_R}$ and the standard deviation s_R must be estimated is the Shapiro-Wilk test defined in the following section.

3.1.1 Shapiro-Wilk test for normality

The Shapiro-Wilk test, calculates a W statistic that tests whether a feature $\mathbf{x} = (x_1, \ldots, x_p)^T$ comes from a normal distribution . Small values of W are evidence of departure from normality. W has a simple interpretation as an approximate measure of the straightness of the normal quantile-quantile ($Q-Q$) probability plot. The W-statistic for a feature is calculated as follows:

$$W = \frac{n\|\mathbf{w^T x}^*\|}{s_R^2} = \frac{(\sum_{i=1}^{n}(w_i x_i^*))^2}{\sum_{i=1}^{n}(x_i - \overline{x_R})^2} \tag{3.1}$$

where $\mathbf{x}^*$ is the ordered data of the original data $\mathbf{x}$, s_R^2 represents the robust variance of a sample (biased estimator of the population variance see Section 3.2.3), $\overline{x_R}$ is the robust sample mean of the feature and $\mathbf{w} = (w_1, \ldots, w_n)$ are weights depending on the number of observations n ($\mathbf{w} = \mathbf{w}(n)$) [15, 16]. W is a measure of the straightness of the normal probability plot.

We apply the Shapiro-Wilk test for all the "cleaned" features and we obtain a total of 14 (11%) normal distributions (see Table 3.1) with a P-value lower than 0.05 (P-value < 0.05).

QQ plots of the normal features are available in appendix C. The feature histograms can be seen in Appendix B and the stem-and-leaf diagrams Section 3.2.2 can also be useful to visualize the distribution. For example, if we look at Stem-and-leaf of the feature $lineVarOfNucl2Avg$ (Section 3.2.2) we can observe that this distribution cannot be normal, in fact values could not be below zero and without such values the distribution looks more like a Poisson distribution.

	feature	*W*	*P-value*	feature	*W*	*P-value*
2	**CorticalB**	0.9968	0.052	nucleoles1	0.9545	1
3	**corticalReaction**	0.9955	0.186	lineVarOfNuclBigAvg	0.9534	1
4	ZS6.GP.avg.if.90	0.9907	0.667	radiusPronBig	0.9531	1
5	OolemmaB	0.9898	0.845	radiusPronTiny	0.9526	1
6	OolemmaA	0.9879	0.922	CorticalRatio	0.9504	1
7	distOfNuclBigToLineAvgRel	0.9879	0.92	radiusPron2Rel	0.9488	1
8	distOfNucl2ToLineAvgRel	0.9856	0.965	posOfPronRel	0.9468	1
9	CorticalA	0.9826	0.987	nucleolesTiny	0.9443	1
10	Ellipse.angle	0.9775	0.998	nucleolesBig	0.9414	1
11	ellDistBetweenRel	0.976	0.999	radiusPron2	0.9376	1
12	proxOfPronCent	0.9738	0.999	lineVarOfNucl1Avg	0.9355	1
13	lineVarOfNucl2Avg	0.9715	1	ZS4.Line.dispersion	0.9355	1
14	dispOfNucl1Avg	0.9712	1	posOfPronSum	0.9259	1
15	radiusPron1Rel	0.9695	1	alphaMin	0.9224	1
16	distOfNucl1ToLineAvgRel	0.9694	1	nucleoles2	0.9194	1
17	OolemmaRatio	0.9691	1	distOfNucl1ToPronLine	0.9059	1
18	proxOfPronRel	0.969	1	distOfNuclBigToPronLine	0.8885	1
19	ellDistCentersRel	0.9675	1	distOfNuclTinyToPronLine	0.8772	1
20	dispOfNuclBigAvg	0.9673	1	diffOfNucl	0.8759	1
21	radiusPron1	0.966	1	distOfNucl2ToPronLine	0.8673	1
22	dispOfNuclTinyAvg	0.9652	1	alpha3D2	0.8587	1
23	sumOfNucl	0.9648	1	ZS5.Cort.react	0.802	1
24	ZS.4bis.Line.position	0.963	1	alphaMax	0.7944	1
25	distOfNuclTinyToLineAvgRel	0.9606	1	ZS.3.Nb.of.nucl	0.7825	1
26	radiusPronTinyRel	0.9602	1	ZS1.Centering	0.7574	1
27	radiusPronBigRel	0.9597	1	alpha3D1	0.7529	1
28	dispOfNucl2Avg	0.9569	1	alphaBetween	0.6669	1
29	lineVarOfNuclTinyAvg	0.9551	1	ZS2.Proximity	0.5001	1
30	nucleoles1	0.9545	1			

Table 3.1: Shapiro-Wilk Test with W-statistic, the non directional alternative hypothesis $H_1 : \rho \neq 0$ is supported at the 0.05 level for bold features

3.1.2 Multivariate normality

Multivariate normality means that the individual features are normal in a univariate sense (see Section 3.1.1) and that their combinations are also normal. Thus, if a variable

is multivariate normal, its components are also normal, but the reverse is not necessarily true. In our case only two features can be considered as normal, so the whole dataset cannot be considered as being normal multivariate.

3.2 Quality of the data and outliers

An outlier is an observation that lies outside the overall pattern of a distribution. Outliers must be viewed within the context of the analysis. They can be of different nature and arise from:

- procedure error (must be eliminated)
- extraordinary event (context must be considered for the cleaning data stage)
- extraordinary observation but ordinary range of values on each feature (context must be considered for the cleaning data stage)

The three cases illustrate the difficulty of an automated exclusion (see Section 3.2.3) or retention of an outlier, an in-context judgment must be made considering the data collection and objectives of the analysis.

3.2.1 "Cleaning" data stage

At this point we consider some criteria for "cleaning" our data from outliers in numerical features. In our case, categorical features are binary, thus univariate outliers methods can not be used with them.

One disadvantage of removing outliers is that the reduction of the sample size can lead to a decrease in statistical significance. However, if we want to adjust a model to our data with, for example, a parametric test such as $t-test$ we had better decreasing the variability of our data. On the other hand, if we include all the data, we increase the probability of having a "distorted" data set. Therefore, invalidating the statistical results.

Although automatic outlier detection algorithms exist, they must be used with care and a visual inspection is absolutely needed. In the following we first explain our visual inspection and then, the automatic outlier detection.

3.2.2 Visual detection

To detect univariate outliers the Stem-and-Leaf plot is chosen. Stem-and-Leaf plot consists of a compact summary of the information contained in a distribution with the original data values. Stem-and-Leaf is a histogram turned on its side and allows us to determine graphically the median and detect outliers. Moreover, detected values can be recovered from the plot. We apply this method to detect features containing outliers and we concluded that only two features contained outliers $lineVarOfNucl2Avg$ and $corticalReaction$. The $lineVarOfNucl2Avg$ is not normally distributed whereas $corticalReaction$ is normally distributed (see Table 3.1), we will see that this property is determinant to chose an automated outlier selection. Features $lineVarOfNucl2Avg$ and $corticalReaction$ are represented in two stem-and-leaf plots, values 44.7 and 43.02, respectively, can be outliers. To detect outliers in these distributions, the algorithms described in Section 3.2.3 are applied.

```
Parameter:  lineVarOfNucl2Avg
All data
Not normally distributed

The decimal point is at the |

 0 | 7933
 2 | 00115668800122345689 9
 4 | 00000001222334456889990000001112223445556688 8
 6 | 000111222223344446678889900001111233334566777788888999
 8 | 0000011222334455566677789999900122234445556688
10 | 00011223688001236788 9
12 | 3567784
14 | 3222
16 |
18 | 08
20 |
22 | 6
24 |
26 |
28 |
30 |
32 |
34 |
36 |
38 |
40 |
42 |
44 | 7   <-   can be an outlier
```

```
Parameter:  corticalReaction
All data
Normaly distributed

The decimal point is at the |

42 | 0   <-   can be an outlier
44 |
46 |
48 |
50 |
52 |
54 |
56 | 9
58 | 0
60 | 8678
62 | 403788
64 | 01124528
66 | 33670011227799
68 | 044458890001345679
70 | 01122234890012457889
72 | 0011224444577789001 3679
74 | 000234445567800000111257
76 | 00001225557778923335677789
78 | 0223366889001245568
80 | 1257790235679
82 | 12237771144678
84 | 0223745
86 | 113570
88 | 77
90 | 4
```

3.2.3 Automated outlier detection

The goal of the algorithms is to select as outliers within a single feature $\mathbf{x_j}$ those observations falling at the outer range of a distribution. Shapiro-Wilk test [15, 16] allows us to decide whether the feature is normally distributed within a confidence interval (see section 3.1.1). If the distribution is considered **normally distributed** Method1 is used otherwise, Method2 is employed.

Method1: robuts mean and standard deviation selection

1. select a constant factor k_{mean}, three for example (99.74% of the observations of a normal distribution will fall within three standard deviations).
2. the conventional arithmetic mean $\bar{x_j}$ and standard deviation of the feature s_j are evaluated.
3. repeat step 2 and 3 without the observations falling out k_{mean} standard deviation of the mean (Equation 3.2) until no new observation is omitted.

$$\frac{\left|x_{ij} - \bar{x_j}\right|}{s} > k_{mean} \tag{3.2}$$

4. The last evaluated mean $\bar{x_j}$ and standard deviation s_j are more robust and the last omitted observations are outliers at level k_{mean} [1]

Method2: median and mad detection

1. evaluate the median absolute deviation mad_j as follows:
   (a) the median $\tilde{x_j}$ of the j-th feature is evaluated.
   (b) for each observation, evaluate the absolute value of the difference between the median and that observation.

[1] k_{mean} is a measure in standard deviation units of how far a score is from the mean for a normally distributed variable, sometimes it is called absolute standard deviation score

(c) the median of step 1b represents the value of mad_j.

(d) mad_j is adjusted by a factor (Equation 3.3) for asymptotically normal consistency (Equation 3.4). Equation 3.5, is the same as Equation 3.2 replacing $\overline{x_j}$ by the median $\tilde{x_j}$ and s_j by mad_j. Sprent describes the procedure as being relatively robust [17].

$$\frac{1}{Phi^{-1}(\frac{3}{4})} = \frac{1}{qnorm(\frac{3}{4})} \approx 1.4826 \tag{3.3}$$

ensures consistency, i.e.,

$$E[mad^*(x_j)] = \sigma_j \qquad \text{for } x_j \text{ distributed as } N(\mu, \sigma^2) \text{ and large } n. \tag{3.4}$$

$$\frac{|p_j - \tilde{p_j}|}{mad_j^*} > k_{median} \tag{3.5}$$

3.2.4 Automated detection comparison

We use the two features selected with outliers in the visual detection section and plot the number of outliers n_{outl} in percent versus k_{mean} (Method1) and k_{median} (Method2) (see Figure 3.1). Method2 is more sensitive than Method1 since the slope n_{outl}/k_{mean} is always greater than n_{outl}/k_{median}. With large k_{median} and k_{mean} the two algorithms have the same effect on this data. However, if we don't know the distribution, it is better to choose Method2 since the first method assumes a normal distribution.

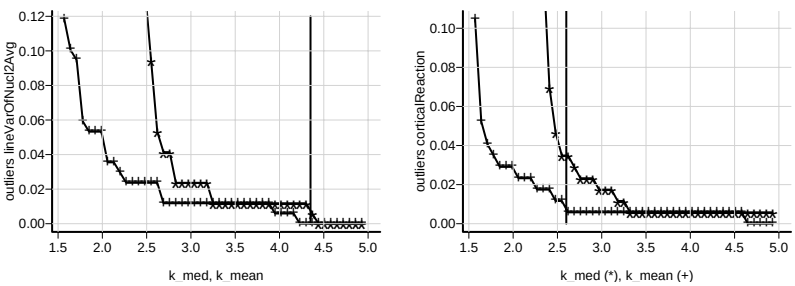


Figure 3.1: Outlier selection of $lineVarOfNucl2Avg$ (left graph) and $corticalReaction$ (right graph) features with Method1 ('+') and Method2 ('*')

Finally, we select the threshold value for each feature and we obtain for the normal feature $k_{mean} = 3.1$ (Equation 3.2) and for the non normal feature $k_{median} = 4.1$ (Equation 3.5). The stem-and-leaf diagrams show the results of our "cleaning stage".

Parameter: **lineVarOfNucl2Avg**
Without outliers
Not normally distributed

The decimal point is at the |

```
 0 | 79
 1 | 33
 2 | 001156688
 3 | 001223456899
 4 | 00000001122233445688999
 5 | 0000001111222344555566888
 6 | 000111222223344446678889 9
 7 | 0000011123333456677778888999
 8 | 00000112223344555666777899999
 9 | 00122234445556688
10 | 00011223688
11 | 0012367889
12 | 356778
13 | 4
14 | 3
15 | 222
16 |
17 |
18 |
19 | 08
20 |
21 |
22 |
23 | 6
```

Parameter: **corticalReaction**
Without outliers
Normaly distributed

The decimal point is at the |

```
56 | 9
58 | 0
60 | 8678
62 | 403788
64 | 01124528
66 | 33670011227799
68 | 044458890001345679
70 | 0112223489012457889
72 | 0011224444577789001 3679
74 | 00023444556780000011 1257
76 | 000012255577789233356 77789
78 | 022366889001245568
80 | 1257790235679
82 | 12237771144678
84 | 0223745
86 | 113570
88 | 77
90 | 4
```

Chapter 4

Individual Classification Power

In this chapter we study the power of each individual feature to classify the **"zygote class"**. For doing this, we measure the degree of association between each feature and the **"zygote class"**. The degree of association is measured differently depending on the nature of the feature: **continuous** or **categorical**.

Continuous data can assume any value within the range of scores that define the limits of a feature. In our case, our continuous features have a relative order and throughout the length of the scale equal differences between measurements correspond to equal differences in the amount of the attribute being measured. For instance, the distance between two sets of points is a continuous feature. On the other hand, categorical data is used in our case to identify mutually exclusive categories. As an example the type of conservation of the zygotes fresh and frozen is a nominal measurement since it is used purely for purpose of identification. Thus, the analysis of the two types of data must be made separately.

4.1 Categorical data

4.1.1 Chi-Square test for independence

The appropriate test to employ for measuring the relationship between the zygote class and a categorical feature is the **Chi-Square test of independence**. This test evalutes the hypothesis that two variables are independent of one another.

A $r \times c = n$ contingency table is constructed with r rows and c columns representing n cases. Observation o_{ij} of the table contains the number of observations falling in the i-th row and the j-th column see Table 4.1.

	variable1	no variable1	row sum
variable2	o_{11}	o_{12}	$o_{[1,.]} = o_{11} + o_{12}$
no variable2	o_{21}	o_{22}	$o_{[2,.]}$
column sum	$o_{[.,1]} = o_{11} + o_{21}$	$o_{[.,2]}$	$o_{tot} = o_{[1,.]} + o_{[2,.]}$

Table 4.1: General model of a $r \times c$ contingency table

	OBS Cryo	Fresh (= no Cryo)	EXP Cryo	Fresh (= no Cryo)
good	34	50	44.04	39.96
bad	74	48	63.96	58.04

Table 4.2: Pearson's Chi-squared test with Yates' continuity correction for the conservation type feature $\chi^2 = 7.3334$, $df = 1$, $p - value = 0.006769$

To apply this test we must make the following assumptions:

1. categorical features for the $r \times c$ contingency table are mutually exclusive categories
2. data represent a random sample, comprised of n-independant observations, selected from the population it represents
3. the expected frequency of each cell in the contingency table is greater or equal to 5

These assumptions are reasonable in our case.

Results: Null versus Alternative Hypotheses

To know if the observed cell frequencies ω_{ij} in the contingency table are different from the expected frequencies ε_{ij} in the underlying population represented by the samples we employ the null and alternative hypotheses under the chi-square test for $r \times c$ table. The null hypothesis is H_0 and the alternative hypothesis is H_1:

$$\begin{array}{rlll} \text{Null Hypothesis:} & H_0: & \omega_{ij} = \varepsilon_{ij} & \text{for all cells} \quad (\equiv \chi = 0) \\ \text{Alternative Hypothesis:} & H_1: & \omega_{ij} \neq \varepsilon_{ij} & \text{for at least one cell} \quad (\equiv \chi \neq 0) \end{array} \tag{4.1}$$

To evaluate the null hypothesis $H_0 : \omega_{ij} = \varepsilon_{ij}$ for all cells Yates' correction for continuity Equation 4.2 is computed. This correction compensates for the fact that a continuous distribution is used to estimate a discrete distribution with the chi-square test [18].

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \left[\frac{(|o_{ij} - e_{ij}| - 0.5)^2}{e_{ij}} \right], \qquad \text{where} \quad e_{ij} = \frac{o_{[i,.]} o_{[.,j]}}{o_{tot}} \tag{4.2}$$

The degrees of freedom employed for the analysis are computed with the number of row and the number of columns, $df = (r-1)(c-1)$. In our particular case we have 2 rows, since we have two classes, and 2 columns ($df = 1$). Critical chi-squared values are $\chi^2_{.05} = \mathbf{3.84}$ and $\chi^2_{.01} = \mathbf{6.63}$.

The results for our two categories are shown in Table 4.2 for the conservation type and in Table 4.3 for the insemination type.

The null hypothesis H_0 can be rejected at both the 0.05 and 0.01 levels for the conservation type (cryo or fresh) $\chi^2 = 7.333$ and $p - value < 0.01$. The significant χ^2 value obtained with the conservation type indicates cryo-preserved zygotes are significantly worse than fresh zygotes. Concerning the insemination type (IVF or ICSI) $\chi^2 = 0.015$ and $p - value = 0.9011$, the null hypothesis can not be rejected rejected. This can also be visually seen on Table 4.2.

	OBS Icsi	Ivf (= no Icsi)	EXP Icsi	Ivf (= no Icsi)
good	48	36	48.93	35.07
bad	72	50	71.07	50.93

Table 4.3: Pearson"'s Chi-squared test with Yates"' continuity correction for the insemination type feature $\chi^2 = 0.0154$, $df = 1$, $p - value = 0.9011$

Measure of association

A measure of association that can be computed is the Pearson's contingency coefficient. The value of the contingency coefficient C is computed with Equation 4.3, where χ^2 is the computed value for the contingency table and n is the total number of observations. The upper limit of the contingency coefficient is C_{max} and can be determined with Equation 4.4, where k is the smaller of the two values of r and c in the contingency table, in our case $k = 2$ and $C_{max} = \sqrt{0.5} \approx 0.71$

$$C = \sqrt{\frac{\chi^2}{\chi^2 + n}} \tag{4.3}$$

$$C_{max} = \sqrt{\frac{k-1}{k}} \tag{4.4}$$

Normalized association can be computed $C/C_{max} \in [-1, 1]$ and is 1 if a perfect association exists between variables. The results for our two categories are $C \approx 0.188$ for the conservation type and $C \approx 0.009$ for the insemination type.

4.1.2 Summary

The assumptions of the **Chi-Square** are fulfilled. **Chi-Square test** evaluates the independence of two variables. We use two categorical features (Icsi/Ivf and Fresh/Cryo) versus the **good zygote** or **bad zygote** class. Our analysis shows that the insemination type does not affect the class of the zygote ($\chi^2 = 0.015$ and $p - value = 0.9011$ not significant). On the other hand, the conservation state of the zygotes is strongly correlated to its success ($\chi^2 = 7.333$ for $p - value < 0.01$ significant). In fact, the difference between those two classes is so large, that in our classification studies we will also analyze zygotes separately depending on whether they are fresh or not.

4.2 Continuous data - Measure of association and correlation

In this section we measure the association between the continuous variables and the **"zygote class"**. In principle, this is traditionally done through the **Pearson product-moment correlation** (Section 4.2.1). However, the hypothesis of this measure are violated by our data. Alternatively, we propose two other measures of association: **Kendall's Tau correlation** (Section 4.2.2) and **Kokmogorov-Smirnov goodness-of-fit** (Section 4.2.3). These tests allow us to have an overview of the importance of each feature. This overview will be summarized in Section 4.2.4.

4.2.1 Pearson product-moment correlation

The most commonly used correlation measure is the **Pearson product-moment correlation** coefficient $\rho \in [-1,1]$. To apply this test we must make some assumptions:

1. the two tested features have a bivariate normal distribution
2. the sample of observations of each feature are randomly selected from the population they represent
3. the relationship between the two features is of equal strength across the whole range of both variables (homoscedasticity)

As was shown in Section 3.1, only two features can be considered to be normal. Therefore, it is difficult to find pairs of bivariate normal variables. Moreover, we are measuring the correlation between a continuous variable and a binary one (the zygote class) which can never be normal. However, although violating the basics assumptions of this association measure, it can still throw light on the degree of importance of a continuous feature compared to the zygote class.

To define the correlation coefficient, first consider the sum of squared values ss_{xx}, ss_{xy}, and ss_{yy}, for two features $\mathbf{x}$ $(\mathbf{x_j})$ and $\mathbf{y}$ $(\mathbf{x_k})$, of a set of n observations (x_i, y_i) about their respective means,

$$\begin{aligned}
ss_{xx} &= \sum_{i=1}^{n}(x_i-\overline{x})^2 &&= \sum_{i=1}^{n} x^2 - n\overline{x}^2 \\
ss_{yy} &= &&= \sum_{i=1}^{n} y^2 - n\overline{y}^2 \\
ss_{xy} &= \sum_{i=1}^{n}(x_i-\overline{x})(y_i-\overline{y}) &&= \sum_{i=1}^{n} xy - n\overline{xy}
\end{aligned}$$

With this notation the **Pearson product-moment correlation** coefficient can be written as follows:

$$\rho = \frac{ss_{xy}}{\sqrt{ss_{xx}ss_{yy}}} \tag{4.5}$$

We compute the **Pearson product-moment correlation** between our features and the class. Considering that for non categorical as for non normal distribution features we cannot conclude the relevance of the test. In fact, even if a functional relationship exists between the class and a feature, $|\rho|$ will be lower than 1 if the normality assumption is violated. To illustrate the bias of this test, we generate an example of a normal distribution $x = N(0,1)$, then we apply the *sign* function $y = sign(x+1)$ to simulate a threshold selection at $x = -1$ (see Figure 4.1). This simulation can perfectly be an idealization of our classification with one feature normally distributed and a threshold at $x = -1$ splitting good zygotes (level 1 in Figure 4.1) and bad (level -1). The **Pearson product-moment correlation** factor in this case is $\rho = 0.694$, as expected it is different from 1.

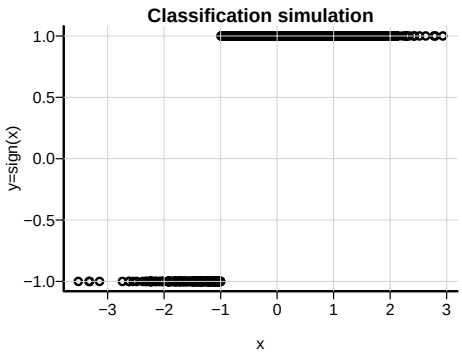


Figure 4.1: Trivial classification simulation with normal an binomial feature

Results: Null versus Alternative Hypotheses To determine if the value of the correlation is large enough to allow to conclude that the underlying population correlation coefficient ρ between the two variables is other than zero, we use an inferential statistical test which is based on the t distribution. The null hypothesis is H_0 and the non directional alternative hypothesis is H_1.

$$\begin{array}{rrl} \text{Null Hypothesis:} & H_0: & \rho = 0 \\ \text{Alternative Hypothesis:} & H_1: & \rho \neq 0 \end{array} \tag{4.6}$$

To evaluate the null hypothesis $H_0 : \rho = 0$ Equation 4.7 is applied employing the t-distribution. Degrees of freedom are nearly constant among features $df = (n-2) > 120(\rightarrow \infty)$ in this case the critical two-tailed value is $\mathbf{t_{.05} = 1.96}$ and $\mathbf{t_{.01} = 2.58}$, in other words it is the standard deviation score above and below which a proportion equivalent to 0.25 ($p = 0.75$) of the cases in the distribution falls.

Our data is represented in Table 4.4. The null hypothesis $H_0 : \rho = 0$ can be rejected with a confidence of 95% if the absolute value of ρ is equal to or greater than $t_{.05} = 1.96$.

$$t = \frac{r\sqrt{n-1}}{\sqrt{1-r^2}} \tag{4.7}$$

In Table 4.4 the bold font indicates those features for which the null hypothesis is rejected at the 95% level of confidence.

4.2.2 Kendall's tau correlation

Definition

As for the **Pearson product-moment correlation**, **Kendall's tau** is a bivariate measure of correlation with rank-order data, this measurement represents the degree of relationship between two variables. The population parameter estimated by the correlation is τ, in our case it will be estimated for each feature and denoted $\tilde{\tau}$. The range of possible values of **Kendall's tau** is defined by the limits -1 to +1 ($\tilde{\tau} \in [-1, 1]$).

As stated above, in order to apply **Kendall's tau** measure of correlation we must quantize our data. Quantizing consists in dividing our features values in to determine ranges. In this analysis the range is divided up into 15 equal intervals. Each interval is created by sampling the observations into 15-subsets of approximately equal size. This is called equal probability quantizing.

1	feature	$\lvert\rho\rvert$	$\lvert t\rvert$	conf interval	feature	$\lvert\rho\rvert$	$\lvert t\rvert$	conf interval
2	**diffOfNucl**	0.303	4.529	[0.17,0.42]	alphaMax	0.076	0.995	[−0.22,0.07]
3	**OolemmaB**	0.241	3.533	[0.11,0.37]	OolemmaRatio	0.069	0.982	[−0.2,0.07]
4	**alphaMin**	0.192	2.537	[0.04,0.33]	nucleolesBig	0.064	0.907	[−0.2,0.07]
5	**CorticalB**	0.191	2.774	[0.06,0.32]	ZS2.Proximity	0.063	0.903	[−0.2,0.07]
6	**OolemmaA**	0.178	2.575	[0.04,0.31]	ZS6.GP.avg.if.90	0.061	0.798	[−0.09,0.21]
7	**dispOfNucl2Avg**	0.17	2.459	[−0.3,−0.03]	distOfNuclBigToPronLine	0.06	0.855	[−0.2,0.08]
8	**nucleoles1**	0.168	2.431	[0.03,0.3]	radiusPronTinyRel	0.058	0.825	[−0.19,0.08]
9	**CorticalA**	0.165	2.383	[0.03,0.3]	lineVarOfNuclBigAvg	0.058	0.829	[−0.19,0.08]
10	**radiusPronBig**	0.153	2.21	[0.02,0.28]	CorticalRatio	0.046	0.65	[−0.18,0.09]
11	**nucleoles2**	0.151	2.182	[−0.28,−0.01]	ZS5.Cort.react	0.044	0.633	[−0.18,0.09]
12	**ZS.4bis.Line.position**	0.149	2.153	[−0.28,−0.01]	distOfNucl1ToPronLine	0.042	0.597	[−0.1,0.18]
13	**distOfNucl2ToPronLine**	0.148	2.131	[−0.28,−0.01]	proxOfPronCent	0.039	0.56	[−0.1,0.18]
14	alpha3D2	0.146	1.924	[0,0.29]	Ellipse.angle	0.039	0.555	[−0.1,0.18]
15	**ellDistCentersRel**	0.141	2.036	[0,0.27]	sumOfNucl	0.038	0.539	[−0.1,0.17]
16	lineVarOfNucl2Avg	0.135	1.921	[−0.27,0]	dispOfNuclTinyAvg	0.036	0.509	[−0.17,0.1]
17	distOfNuclBigToLineAvgRel	0.135	1.937	[−0.27,0]	posOfPronRel	0.034	0.49	[−0.17,0.1]
18	radiusPron1	0.132	1.894	[−0.01,0.26]	radiusPron2Rel	0.032	0.462	[−0.17,0.11]
19	nucleolesTiny	0.132	1.898	[−0.01,0.26]	ZS.3.Nb.of.nucl	0.028	0.395	[−0.11,0.16]
20	alpha3D1	0.131	1.721	[−0.02,0.28]	ellDistBetweenRel	0.027	0.385	[−0.11,0.16]
21	alphaBetween	0.127	1.671	[−0.02,0.27]	lineVarOfNuclTinyAvg	0.024	0.337	[−0.11,0.16]
22	dispOfNuclBigAvg	0.125	1.801	[−0.26,0.01]	radiusPron1Rel	0.015	0.213	[−0.15,0.12]
23	distOfNucl1ToLineAvgRel	0.121	1.732	[−0.25,0.02]	radiusPronBigRel	0.012	0.172	[−0.13,0.15]
24	distOfNucl2ToLineAvgRel	0.115	1.647	[−0.25,0.02]	posOfPronSum	0.007	0.094	[−0.13,0.14]
25	distOfNuclTinyToLineAvgRel	0.106	1.521	[−0.24,0.03]	ZS1.Centering	0.005	0.077	[−0.13,0.14]
26	radiusPronTiny	0.083	1.19	[−0.05,0.22]	corticalReaction	0.004	0.064	[−0.13,0.14]
27	radiusPron2	0.082	1.166	[−0.06,0.22]	distOfNuclTinyToPronLine	0.002	0.027	[−0.14,0.14]
28	lineVarOfNucl1Avg	0.079	1.121	[−0.06,0.21]	dispOfNucl1Avg	0.002	0.024	[−0.14,0.14]
29	ZS4.Line.dispersion	0.079	1.121	[−0.06,0.21]	proxOfPronRel	0.002	0.033	[−0.13,0.14]
30	alphaMax	0.076	0.995	[−0.22,0.07]				

Table 4.4: Pearson product-moment correlation of all features with the *class* feature, the non directional alternative hypothesis $H_1 : \rho \neq 0$ is supported at the 0.05 level for bold features

To define the correlation coefficient, first consider a ranked feature $\mathbf{x} = (x_1, \ldots, x_p)^T$ and the class ranked feature $\mathbf{y} = (y_1, \ldots, y_p)^T$, where $x_i \in \{1, 2, \ldots, 6\}$ and $y_i \in \{1, 2\}$. If the sign of the difference $x_i - x_j$ is the same as the sign of the difference $y_i - y_j$, where i and j represent respectively the i-th and j-th observation, then a pair of ranks is said to be **concordant**, otherwise it is said to be **discordant**.

To compute the value of $\tilde{\tau}$ Equation 4.8 is applied. Where, n_C is the number of concordant pairs of ranks n_D the number of discordant and $n/2(n-1)$ the total number of possible pairs of ranks.

$$\tilde{\tau} = \frac{2\,(n_C - n_D)}{(n-1)} \tag{4.8}$$

Results: Null versus Alternative Hypotheses

To determine if the value of the correlation is large enough to allow to conclude that underlying population correlation coefficient τ between the two variables is other than zero, we use an inferential statistical test. The null hypothesis is H_0 and the non directional alternative hypothesis is H_1.

$$\begin{aligned} \text{Null Hypothesis:} \qquad & H_0 : \quad \tau = 0 \\ \text{Alternative Hypothesis:} \qquad & H_1 : \quad \tau \neq 0 \end{aligned} \tag{4.9}$$

To evaluate the null hypothesis $H_0 : \tau = 0$ Equation 4.8 can be applied among features in this case the critical two-tailed value for $n > 40$ is $\tilde{\tau}_{.05} = \mathbf{0.218}$ and $\tilde{\tau}_{.01} = \mathbf{0.285}$. Or when $n > 10$ the normal distribution provides an excellent approximation of the sampling distribution of τ [19]. The normal approximation can made with Equation 4.10

The critical two-tailed value for a normal distribution is $\mathbf{z}_{.05} = \mathbf{1.96}$ and $\mathbf{z}_{.01} = \mathbf{2.58}$. The null hypothesis $H_0 : \tau = 0$ is rejected if the absolute value of z is equal to or greater

than $\mathbf{z_{.05} = 1.96}$. In Table 4.5 the bold features reject the null hypothesis at the 0.05 level of confidence.

$$z = \frac{3\tilde{\tau}\sqrt{n(n-1)}}{\sqrt{2(2n+5)}} \tag{4.10}$$

	feature	$\lvert\tilde{\tau}\rvert$	$\lvert z\rvert$	feature	$\lvert\tilde{\tau}\rvert$	$\lvert z\rvert$
2	**ZS.3.Nb.of.nucl**	0.302	5.833	distOfNuclTinyToLineAvgRel	0.079	1.519
3	**diffOfNucl**	0.3	5.798	ZS6.GP.avg.if.90	0.074	1.431
4	**ZS2.Proximity**	0.287	5.541	alpha3D1	0.068	1.316
5	**ZS1.Centering**	0.278	5.368	radiusPron2	0.065	1.249
6	**nucleoles2**	0.185	3.568	OolemmaA	0.06	1.151
7	**nucleolesBig**	0.182	3.514	CorticalA	0.054	1.048
8	**alphaMin**	0.177	3.415	lineVarOfNuclTinyAvg	0.047	0.898
9	**OolemmaB**	0.175	3.368	radiusPronTinyRel	0.046	0.889
10	**ZS.4bis.Line.position**	0.169	3.271	dispOfNucl2Avg	0.044	0.858
11	**distOfNucl1ToLineAvgRel**	0.161	3.116	radiusPronBigRel	0.043	0.827
12	**distOfNucl2ToPronLine**	0.157	3.028	proxOfPronRel	0.035	0.667
13	**distOfNuclBigToLineAvgRel**	0.157	3.035	posOfPronSum	0.034	0.648
14	**CorticalB**	0.156	3.014	alphaMax	0.03	0.575
15	**radiusPronTiny**	0.154	2.968	CorticalRatio	0.029	0.567
16	**sumOfNucl**	0.141	2.726	distOfNuclTinyToPronLine	0.026	0.495
17	**alpha3D2**	0.139	2.686	Ellipse.angle	0.022	0.416
18	**dispOfNuclBigAvg**	0.132	2.553	radiusPron1	0.016	0.311
19	**ellDistCentersRel**	0.124	2.393	posOfPronRel	0.015	0.291
20	**lineVarOfNuclBigAvg**	0.114	2.205	radiusPron2Rel	0.012	0.236
21	**distOfNuclBigToPronLine**	0.109	2.103	ellDistBetweenRel	0.01	0.184
22	**radiusPronBig**	0.108	2.079	dispOfNucl1Avg	0.008	0.153
23	**lineVarOfNucl2Avg**	0.105	2.032	proxOfPronCent	0.006	0.121
24	distOfNucl2ToLineAvgRel	0.101	1.944	nucleolesTiny	0.006	0.117
25	OolemmaRatio	0.101	1.948	dispOfNuclTinyAvg	0.006	0.115
26	corticalReaction	0.1	1.935	radiusPron1Rel	0.005	0.098
27	alphaBetween	0.099	1.92	nucleoles1	0.004	0.085
28	ZS5.Cort.react	0.094	1.815	lineVarOfNucl1Avg	0.002	0.04
29	distOfNucl1ToPronLine	0.081	1.572	ZS4.Line.dispersion	0.002	0.04
30	distOfNuclTinyToLineAvgRel	0.079	1.519			

Table 4.5: Kendall's tau for six ranks segmented features, the non directional alternative hypothesis $H_1 : \tau \neq 0$ is supported at the 0.05 level for bold features

In order to apply **Kendall's tau test** which measures the degree of agreement between two sets of ranks, our features must be ranked with k labels. A segmentation process must be applied in order to have the same number of k labels in each feature.

4.2.3 Kolmogorov-Smirnov goodness-of-fit test

We expect that some features are drawn from two different distributions depending on the zygote class (see Section 3.1), thus we would like to compare the distribution between classes. If a difference is noticed, then there exists a significant relationship between that feature and the zygote class.

Kolmogorov-Smirnov test is employed to compare the cumulative frequency distributions of two independent samples. To apply this test we must make some assumptions:

1. the sample of observations in each class are randomly selected from the population they represent
2. all of the samples of observations in each class are independent of one other
3. the scale of measurement is at least ordinal

As can be seen our data fully meet these assuptions.

To apply the **Kolmogorov-Smirnov test** we calculate the cumulative frequency (normalized by the sample size) of the observations for the **"good zygote"** sample F_{good} and for the **"bad zygote"** sample F_{bad}. The D-statistic is defined as the greatest

distance between the observed and expected cumulative probability distribution at any point. To compute the value of D Equation 4.11 is applied, where j represents the j-th cumulated frequency over the two samples.

$$D = max_j \left(F_{good,j} - F_{bad,j}\right) \tag{4.11}$$

This method is sensitive to outliers because we take the maximum value and it can be biased by outliers, so the stage of outliers detection (Section 3.2.1) is essential.

Results: Null versus Alternative Hypotheses To determine wether the distribution of the "bad zygote" sample derived from its population is consistent with the distribution of "good zygote", the following two hypothesis are considered:

$$\begin{aligned} \text{Null Hypothesis:} \quad & H_0: \quad F_{good} = F_{bad} \\ \text{Alternative Hypothesis:} \quad & H_1: \quad F_{good} \neq F_{bad} \end{aligned} \tag{4.12}$$

To conclude the goodness of fit we evaluate the null hypothesis $H_0 : F_{good} = F_{bad}$ Equation 4.11 is applied among features. If the calculated D-statistic is greater than the critical one (p < 0.05), then the hypothesis that both distribution are equal should be rejected. In Table 4.6 the bold features reject the null hypothesis.

	feature	D	$P-$value	feature	D	$P-$value
2	**diffOfNucl**	0.27	0.002	nucleolesTiny	0.136	0.322
3	**ZS.4bis.Line.position**	0.252	0.004	ellDistCentersRel	0.13	0.375
4	**distOfNuclBigToPronLine**	0.25	0.004	OolemmaRatio	0.129	0.386
5	**OolemmaB**	0.248	0.004	proxOfPronCent	0.126	0.412
6	**alpha3D2**	0.245	0.013	lineVarOfNucl1Avg	0.125	0.424
7	**alphaMin**	0.228	0.021	ZS4.Line.dispersion	0.125	0.424
8	**nucleoles2**	0.223	0.015	radiusPronBig	0.119	0.488
9	**distOfNuclBigToLineAvgRel**	0.218	0.018	dispOfNucl2Avg	0.118	0.499
10	**distOfNucl1ToLineAvgRel**	0.209	0.027	radiusPron2	0.113	0.558
11	**CorticalB**	0.208	0.028	distOfNucl1ToPronLine	0.111	0.576
12	**lineVarOfNucl2Avg**	0.206	0.034	posOfPronRel	0.104	0.659
13	**distOfNucl2ToPronLine**	0.201	0.037	radiusPronTiny	0.103	0.666
14	**CorticalA**	0.196	0.045	dispOfNuclTinyAvg	0.101	0.699
15	distOfNucl2ToLineAvgRel	0.181	0.079	posOfPronSum	0.101	0.693
16	distOfNuclTinyToLineAvgRel	0.166	0.131	dispOfNucl1Avg	0.097	0.746
17	nucleolesBig	0.165	0.135	proxOfPronRel	0.09	0.818
18	nucleoles1	0.164	0.142	ellDistBetweenRel	0.082	0.893
19	alphaBetween	0.162	0.22	distOfNuclTinyToPronLine	0.076	0.939
20	lineVarOfNuclTinyAvg	0.157	0.182	Ellipse.angle	0.075	0.942
21	alpha3D1	0.155	0.264	sumOfNucl	0.058	0.996
22	ZS6.GP.avg.if.90	0.153	0.246	radiusPronTinyRel	0.055	0.998
23	dispOfNuclBigAvg	0.148	0.231	ZS2.Proximity	0.052	0.999
24	radiusPron1	0.148	0.229	ZS5.Cort.react	0.05	1
25	alphaMax	0.147	0.328	radiusPron2Rel	0.042	1
26	OolemmaA	0.145	0.251	radiusPron1Rel	0.032	1
27	corticalReaction	0.143	0.262	radiusPronBigRel	0.032	1
28	lineVarOfNuclBigAvg	0.138	0.315	ZS1.Centering	0.029	1
29	CorticalRatio	0.137	0.308	ZS.3.Nb.of.nucl	0.02	1
30	nucleolesTiny	0.136	0.322			

Table 4.6: Kolmogorov-Smirnov goodness-of-fit test, in bold those features for which the D-statistic is greater than the critical one

4.2.4 Correlation Summary

Severals tests for the measure of association and correlation were considered since the assumptions of some of them are violated. A summary of important features is shown in Table 4.7.

In Table 4.8 a score is calculated considering the order of each feature, the higher the feature is, the higher is the score. We apply this scoring to each test **Pearson**, **Kendall** and **Kolmogorov** and then, we add the three scores, finally we normalize the result. Three other columns are marked with a cross if the feature were significant in

	pearson		kedall		kolmogorov	
2	feature	$\lvert\rho\rvert$	feature	$\lvert\tau\rvert$	feature	D
3	diffOfNucl	0.303	ZS.3.Nb.of.nucl	0.302	diffOfNucl	0.27
4	OolemmaB	0.241	diffOfNucl	0.3	ZS.4bis.Line.position	0.252
5	alphaMin	0.192	ZS2.Proximity	0.287	distOfNuclBigToPronLine	0.25
6	CorticalB	0.191	ZS1.Centering	0.278	OolemmaB	0.248
7	OolemmaA	0.178	nucleoles2	0.185	alpha3D2	0.245
8	dispOfNucl2Avg	0.17	nucleolesBig	0.182	alphaMin	0.228
9	nucleoles1	0.168	alphaMin	0.177	nucleoles2	0.223
10	CorticalA	0.165	OolemmaB	0.175	distOfNuclBigToLineAvgRel	0.218
11	radiusPronBig	0.153	ZS.4bis.Line.position	0.169	distOfNucl1ToLineAvgRel	0.209
12	nucleoles2	0.151	distOfNucl1ToLineAvgRel	0.161	CorticalB	0.208
13	ZS.4bis.Line.position	0.149	distOfNucl2ToPronLine	0.157	lineVarOfNucl2Avg	0.206
14	distOfNucl2ToPronLine	0.148	distOfNuclBigToLineAvgRel	0.157	distOfNucl2ToPronLine	0.201
15	alpha3D2	0.146	CorticalB	0.156	CorticalA	0.196
16	ellDistCentersRel	0.141	radiusPronTiny	0.154	distOfNucl2ToLineAvgRel	0.181
17	lineVarOfNucl2Avg	0.135	sumOfNucl	0.141	distOfNuclTinyToLineAvgRel	0.166
18	distOfNuclBigToLineAvgRel	0.135	alpha3D2	0.139	nucleolesBig	0.165
19	radiusPron1	0.132	dispOfNuclBigAvg	0.132	nucleoles1	0.164
20	nucleolesTiny	0.132	ellDistCentersRel	0.124	alphaBetween	0.162
21	alpha3D1	0.131	lineVarOfNuclBigAvg	0.114	lineVarOfNuclTinyAvg	0.157
22	alphaBetween	0.127	distOfNuclBigToPronLine	0.109	alpha3D1	0.155

Table 4.7: Summary of measure of association and correlation with three tests in the order of descending importance

their corresponding test. We chose a total of seven features (crosses throughout the three columns) and we consider them significant in respect to the class **good** or **bad**. We keep in mind these seven features for the subsequent analysis.

	summary					summary				
2	feature	score	pearson	kedall	kolmog.	feature	score	pearson	kedall	kolmog.
3	diffOfNucl	1	x	x	x	ZS2.Proximity	0.726	–	x	–
4	OolemmaB	0.966	x	x	x	OolemmaRatio	0.726	–	–	–
5	alphaMin	0.959	x	x	x	alphaMax	0.696	–	–	–
6	ZS.4bis.Line.position	0.939	x	x	x	radiusPron2	0.693	–	–	–
7	nucleoles2	0.939	x	x	x	nucleolesTiny	0.682	–	–	–
8	CorticalB	0.922	x	x	x	lineVarOfNuclTinyAvg	0.669	–	–	–
9	alpha3D2	0.899	–	x	x	ZS.3.Nb.of.nucl	0.666	–	x	–
10	distOfNucl2ToPronLine	0.895	x	x	x	corticalReaction	0.662	–	–	–
11	distOfNuclBigToLineAvgRel	0.892	–	x	x	distOfNucl1ToPronLine	0.659	–	–	–
12	distOfNucl1ToLineAvgRel	0.875	–	x	x	sumOfNucl	0.659	–	x	–
13	lineVarOfNucl2Avg	0.851	–	x	x	CorticalRatio	0.652	–	–	–
14	nucleolesBig	0.834	–	x	–	ZS1.Centering	0.639	–	x	–
15	CorticalA	0.828	x	–	x	lineVarOfNucl1Avg	0.625	–	–	–
16	distOfNuclBigToPronLine	0.821	–	x	x	ZS5.Cort.react	0.622	–	–	–
17	dispOfNuclBigAvg	0.811	–	x	–	ZS4.Line.dispersion	0.615	–	–	–
18	distOfNucl2ToLineAvgRel	0.811	–	–	–	radiusPronTinyRel	0.608	–	–	–
19	ellDistCentersRel	0.804	x	x	–	proxOfPronCent	0.601	–	–	–
20	OolemmaA	0.801	x	–	–	posOfPronRel	0.578	–	–	–
21	alphaBetween	0.797	–	–	–	Ellipse.angle	0.568	–	–	–
22	radiusPronBig	0.794	x	x	–	posOfPronSum	0.564	–	–	–
23	distOfNuclTinyToLineAvgRel	0.784	–	–	–	dispOfNuclTinyAvg	0.554	–	–	–
24	alpha3D1	0.777	–	–	–	proxOfPronRel	0.544	–	–	–
25	nucleoles1	0.75	x	–	–	ellDistBetweenRel	0.541	–	–	–
26	radiusPronTiny	0.747	–	x	–	radiusPronBigRel	0.534	–	–	–
27	dispOfNucl2Avg	0.747	x	–	–	distOfNuclTinyToPronLine	0.53	–	–	–
28	lineVarOfNuclBigAvg	0.736	–	x	–	radiusPron2Rel	0.527	–	–	–
29	ZS6.GP.avg.if.90	0.73	–	–	–	dispOfNucl1Avg	0.517	–	–	–
30	radiusPron1	0.726	–	–	–	radiusPron1Rel	0.49	–	–	–

Table 4.8: Columns are marked with a cross if the feature is significant for each test

4.2.5 Principal Components Analysis

The purpose of **principal components analysis (PCA)** is to find a smaller group of underlying variables (p) that describe the data. These new variables (r) are uncorrelated linear combinations of the original variables ($p > r$). The new variables rarely lead to a greater understanding of the data, it is difficult to find a meaning for them. On the other hand, the smaller new group of variables allows us to

- provide a relevant set of features for a classifier
- reduce redundancy;

None of these two applications are relevant to us in this case. We will make use of **PCA** with a different goal: measure the contribution of each feature to the overall variance. In this way, if a feature has a strong contribution to the overall variance, it means either that that feature is very noisy or that it is strongly related to an underlying physical process (helpfully the zygote class).

PCA is an eigenanalysis, it can be obtained using the **covariance (C)** or the **correlation (R)** matrix (see Equations 4.13a and 4.13b). We want to give all features the same variation by standardizing, so we will make use of the correlation matrix.

We perform an **eigenvector decomposition** of **R** defined by Equation 4.13b. Principal components can be determined using eigenvectors as coefficients in the linear combination of the variables. Let $\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_p})$ be our set of original features, c_{xy} represent the covariance between features $\mathbf{x}$ $(\mathbf{x_j})$ and $\mathbf{y}$ $(\mathbf{x_k})$ (see Equation 4.13a). We use **R** of rank r to calculate an orthogonal basis by finding real positive **eigenvalues** $\lambda_1, \ldots, \lambda_r$ (**R** is symmetric positive definite) and eigenvectors $\mathbf{c_i}$. Eigenvectors and eigenvalues are the solutions of Equation 4.15. We order the eigenvectors in the order of descending eigenvalues, it is equivalent to the order of largest correlation.

$$\mathbf{C} = \begin{pmatrix} c_1 & \ldots & c_{1r} \\ \vdots & \ddots & \vdots \\ c_{r1} & \ldots & c_r \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} 1 & \ldots & r_{1r} = \frac{c_{1r}}{c_1 c_r} \\ \vdots & \ddots & \vdots \\ r_{r1} = \frac{c_{r1}}{c_1 c_r} & \ldots & 1 \end{pmatrix} \tag{4.13a}$$

$$c_{xy} = \frac{ss_{xy}}{(n-1)} = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}), \qquad r_{xy} = \frac{c_{xy}}{c_{xx}c_{yy}}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y}) \tag{4.13b}$$

$$\mathbf{R}\mathbf{c_i} = \lambda_i \mathbf{c_i}, \qquad (\mathbf{R} - \lambda_\mathbf{i}\mathbf{I})\,\mathbf{c_i} = 0 \tag{4.14}$$

$$\text{non trivial solutions for} \quad |\mathbf{R} - \lambda\mathbf{I}| \tag{4.15}$$

Results

In this section we first study the contribution of each principal component to the overall variance, then we study the contribution of each feature within each component.

The first principal component ($component1$) is the component that explains the greatest part of the variation, then after the first component the second component $component2$ explains the greatest part of the variation, and so on. A bar plot of the relative variance of the 37 first components is shown in Figure 4.2. This relative variance of each component represents the contribution of each component to the overall variance. The decrease in contributions from the principal components is relatively low, to obtain 99% of the overall variance, 37 first components must be used; with ten components we have only 52% of the overall variance. Variance of components 50 (r) to 56 (p) are equal to zero, so the underlying variables that describe the data have a smaller dimension ($r < p$).

In Table 4.9 you can find the absolute value of the first three ordered components ($c_i, i \in \{1,2,3\}$). In bold font are those features for which the power to classify the **"zygote class"** tests are significant at 0.05 level of confidence (see Section 4.2.4). These features are also marked with a circle in the 2-dimensional representation of the data (Figures 4.3, 4.4, 4.5). It can be seen that significant variables are widespread

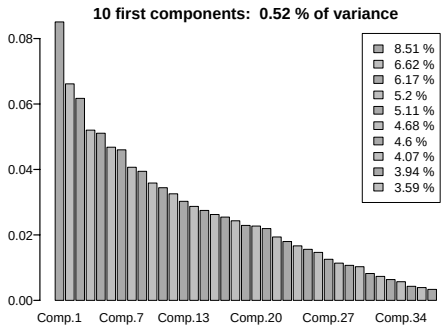


Figure 4.2: Relative variance of the 37 first components, 99% of the overall variance

on the PCA axis. This means that they are not too much correlated among them and, therefore, they bring enough information about the data set.

Bold features in Table 4.9 features are scattered across columns, that illustrates the fact that noisy features has a strong contribution to the overall variance and that a classification with these components is not recommended. This components can still throw light on the degree of correlation of features for which the power to classify the **"zygote class"** tests were significant.

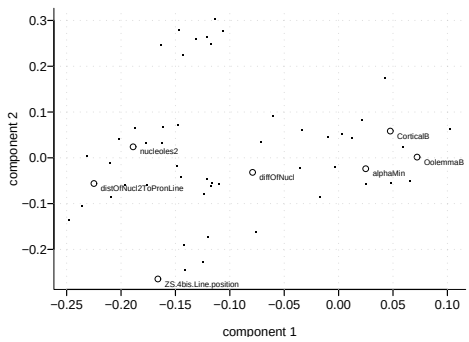


Figure 4.3: $component1$ function of $component2$

Each principal component is a linear combination of the initial data involving difficulties to find a meaning with a set of them, especially in our case, some explanation of the importance of each feature may be necessary to get the methods adopted, furthermore to provide a reasonable summary of the data nearly all components must be used. However, this test was useful to measure the contribution of each feature to the overall variance. PCA is a popular technique in pattern recognition. However, PCA is not optimized for class separability. An alternative is the **linear discriminant analysis** (see Section 5.3).

	feature	$\lvert component1 \rvert$	feature	$\lvert component2 \rvert$	feature	$\lvert component3 \rvert$
2	distOfNucl1ToPronLine	0.248	radiusPronTiny	0.304	proxOfPronCent	0.295
3	distOfNuclBigToPronLine	0.236	radiusPron2	0.278	proxOfPronRel	0.269
4	sumOfNucl	0.232	radiusPronBig	0.278	posOfPronRel	0.229
5	**distOfNucl2ToPronLine**	0.225	**ZS.4bis.Line.position**	0.265	radiusPronTinyRel	0.211
6	nucleoles1	0.21	radiusPronTinyRel	0.265	ZS1.Centering	0.204
7	distOfNuclTinyToPronLine	0.209	radiusPron1	0.261	posOfPronSum	0.203
8	ZS.3.Nb.of.nucl	0.202	radiusPron2Rel	0.248	radiusPron2Rel	0.197
9	nucleolesBig	0.197	radiusPronBigRel	0.246	ZS.3.Nb.of.nucl	0.184
10	**nucleoles2**	0.189	distOfNucl1ToLineAvgRel	0.244	sumOfNucl	0.184
11	dispOfNucl1Avg	0.187	distOfNuclTinyToLineAvgRel	0.227	nucleoles1	0.183
12	dispOfNuclBigAvg	0.177	radiusPron1Rel	0.224	lineVarOfNucl1Avg	0.183
13	ZS4.Line.dispersion	0.176	distOfNuclBigToLineAvgRel	0.19	ZS2.Proximity	0.179
14	**ZS.4bis.Line.position**	0.166	ZS2.Proximity	0.174	distOfNuclTinyToLineAvgRel	0.163
15	radiusPronBigRel	0.163	distOfNucl2ToLineAvgRel	0.172	nucleolesBig	0.162
16	dispOfNuclTinyAvg	0.162	proxOfPronRel	0.162	**ZS.4bis.Line.position**	0.155
17	dispOfNucl2Avg	0.162	distOfNucl1ToPronLine	0.135	OolemmaA	0.148
18	nucleolesTiny	0.148	distOfNuclBigToPronLine	0.106	radiusPronTiny	0.146
19	lineVarOfNucl1Avg	0.148	alphaMax	0.091	ZS4.Line.dispersion	0.143
20	radiusPronBig	0.146	ellDistCentersRel	0.086	distOfNucl2ToLineAvgRel	0.142
21	lineVarOfNuclBigAvg	0.145	distOfNuclTinyToPronLine	0.086	radiusPron2	0.141
22	radiusPron1Rel	0.143	ellDistBetweenRel	0.083	radiusPronBigRel	0.137
23	distOfNuclBigToLineAvgRel	0.142	lineVarOfNucl2Avg	0.078	ZS5.Cort.react	0.129
24	distOfNucl1ToLineAvgRel	0.141	nucleolesTiny	0.071	**nucleoles2**	0.128
25	radiusPron1	0.132	dispOfNuclTinyAvg	0.067	radiusPron1Rel	0.127
26	distOfNuclTinyToLineAvgRel	0.124	dispOfNucl1Avg	0.065	lineVarOfNuclBigAvg	0.126
27	lineVarOfNucl2Avg	0.124	ZS1.Centering	0.062	dispOfNucl1Avg	0.125
28	radiusPronTinyRel	0.121	posOfPronRel	0.061	alphaMax	0.121
29	lineVarOfNuclTinyAvg	0.121	ZS6.GP.avg.if.90	0.06	corticalReaction	0.119
30	distOfNucl2ToLineAvgRel	0.12	ZS4.Line.dispersion	0.059	ellDistBetweenRel	0.116
31	radiusPron2Rel	0.118	nucleolesBig	0.058	nucleolesTiny	0.111
32	posOfPronRel	0.117	**CorticalB**	0.058	dispOfNuclBigAvg	0.11
33	proxOfPronCent	0.116	alpha3D2	0.058	alphaBetween	0.107
34	radiusPronTiny	0.113	posOfPronSum	0.058	alpha3D1	0.105
35	posOfPronSum	0.11	**distOfNucl2ToPronLine**	0.056	ZS6.GP.avg.if.90	0.104
36	radiusPron2	0.107	proxOfPronCent	0.055	distOfNucl1ToLineAvgRel	0.102
37	ZS1.Centering	0.103	alphaBetween	0.054	**diffOfNucl**	0.098
38	**diffOfNucl**	0.079	corticalReaction	0.052	**OolemmaB**	0.097
39	proxOfPronRel	0.076	alpha3D1	0.05	lineVarOfNuclTinyAvg	0.089
40	**OolemmaB**	0.072	lineVarOfNuclTinyAvg	0.047	ellDistCentersRel	0.083
41	OolemmaRatio	0.072	OolemmaA	0.046	alpha3D2	0.076
42	alpha3D1	0.065	CorticalA	0.044	distOfNuclBigToLineAvgRel	0.073
43	alphaMax	0.06	lineVarOfNuclBigAvg	0.043	distOfNuclBigToPronLine	0.07
44	Ellipse.angle	0.059	ZS.3.Nb.of.nucl	0.041	distOfNuclTinyToPronLine	0.062
45	alphaBetween	0.048	OolemmaRatio	0.034	radiusPronBig	0.053
46	**CorticalB**	0.047	dispOfNucl2Avg	0.033	radiusPron1	0.046
47	ZS2.Proximity	0.043	**diffOfNucl**	0.032	**CorticalB**	0.042
48	CorticalRatio	0.035	dispOfNuclBigAvg	0.031	distOfNucl1ToPronLine	0.036
49	ZS6.GP.avg.if.90	0.034	**nucleoles2**	0.024	CorticalRatio	0.034
50	alpha3D2	0.025	**alphaMin**	0.024	**distOfNucl2ToPronLine**	0.03
51	**alphaMin**	0.025	Ellipse.angle	0.024	OolemmaRatio	0.029
52	ellDistBetweenRel	0.021	CorticalRatio	0.022	lineVarOfNucl2Avg	0.029
53	ellDistCentersRel	0.017	ZS5.Cort.react	0.02	CorticalA	0.011
54	CorticalA	0.012	lineVarOfNucl1Avg	0.017	dispOfNuclTinyAvg	0.006
55	OolemmaA	0.01	nucleoles1	0.011	**alphaMin**	0.006
56	corticalReaction	0.003	sumOfNucl	0.004	dispOfNucl2Avg	0.004
57	ZS5.Cort.react	0.003	**OolemmaB**	0.002	Ellipse.angle	0.001

Table 4.9: Absolute value of the first three ordered components, in bold font are those features for which the power to classify the zygote class are significant

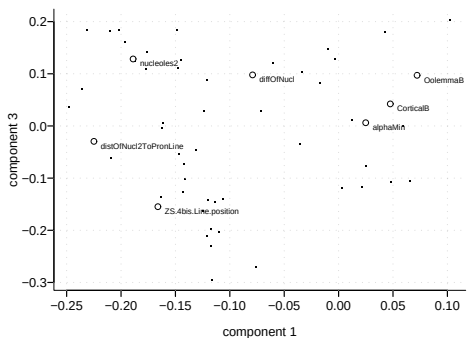


Figure 4.4: $component1$ function of $component3$

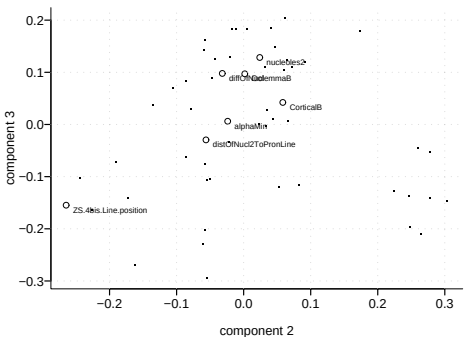


Figure 4.5: $component2$ function of $component3$

Chapter 5

Supervised Classifiers

We start this chapter with some definitions that will be useful to assess the quality of the classifier at hard (Section 5.1). Then we explain two different classification strategies that will be used throughout the chapter. Finally, we show the essential theory and the results obtained with a full battery of classifiers.

Our goal in this chapter is to predict the implantation potential of the zygotes $y_i \in \{1,2\}$ based on a i-th observation $\mathbf{x_i}$. Each zygote potential data are divided in three classes, zygotes leading to pregnancy (twins) zygotes no leading to pregnancy and uncertain zygotes (see Chapter 2). Only the first two classes are going to be used in this chapter.

Two datasets are compared, the first one is the **PNS** zygote scoring data, zygotes were visually scored from one to three for proximity, orientation and centering of the pronuclei, cytoplasmic halo, and number and polarisation of the nucleoli. From these individual scores, a cumulated pronuclear score labeled **PNS** is calculated. The second dataset is the data collected with the *ImageJ Plug-In*, pronuclear measurements were made (see Chapter 2) and it is labeled **IJPM**. Furthermore each dataset can also be divided in fresh and frozen-thawed zygotes labeled **fresh** and **cryo**, respectively (see Section 2.1.2).

5.1 Classification accuracy

To assess the performance of our classifiers, the **error rate** will be computed splitting our data in two parts, the training set and the test set. Several procedures are available to compute the error rate. Three of them are **k-fold cross-validation** (5.1), **bootstrap** (5.1) and **holdout estimate**. The first two methods are used throughout this chapter. In fact, k-fold cross-validation technique is superior than **holdout estimate**, giving reduced bias for small data sets [20]. Furthermore, k-fold cross-validation (if $k \neq n$) is also superior in terms of variance than **leave-one-out cross-validation** ($k = n$) procedure. Indeed if the prediction rule is unstable the leave-one-out procedure will have high variance, the sets being too similar to the full data set. Concerning bootstrap procedure, it has proved superior to many other techniques [14] making an efficient use of the data also compared to the holdout estimate procedure using only a little part of the data in the training process.

Before computing errors rates let us introduce some notation. Let the training data be $\mathbf{Y} = \{\mathbf{y}_i, i = 1,\ldots,m\}$ and let $\omega = \{\omega_i, i = 1,\ldots,m\}$ be the true categorical class

labels. Now consider our classifier built with the training data $\mathbf{y}$ and its corresponding class labels $\mathbf{o} = \{o_i = 1, \ldots, m\}$ assigned with a given classifier. Now let $Q(\omega_i, o_i)$ be the loss function

$$Q(\omega_i, o_i) = \begin{cases} 0 & \text{if } \omega_i = o_i \quad \text{(correct classification)} \\ 1 & \text{if } \omega_i \neq o_i \quad \text{(misclassified)} \end{cases} \tag{5.1}$$

Let e^A be the **apparent error rate** for a given classifier, necessary to calculate the bootstrap error rate. The apparent error rate is obtained by cumulating the errors through the training data.

$$e^A = \frac{1}{m} \sum_{i=1}^{m} Q(\omega_i, o_i) \tag{5.2}$$

This error rate is optimistically biased, in fact we consider the same data to measure the error and to build the classifier. The following sections summarize the k-fold cross-validation and the bootstrap algorithms.

k-fold cross-validation

k-fold cross-validation is an estimator method based on resampling. k-fold cross-validation algorithm is created by sampling the observations into k-subsets of approximately equal size, then classify k times, each time leaving out one of the subsets from training, but using only the omitted subset to compute the misclassification error rate. It is implemented as follows

1. Sample the data into k-subsets of approximately equal size $\mathbf{Y} = \{\mathbf{y}_i, i = 1, \ldots, k\})$
2. Generate the classifier with a data set each time leaving out one of the k-subsets from training $\mathbf{Y}_j^C$, where $\mathbf{Y}_j$ is the k-th subset and $\mathbf{Y}_j^C$ is its complementary subset

$$\begin{aligned} \mathbf{Y}_j &= \{\mathbf{y}_i, i = 1, \ldots, \left\lfloor \frac{1}{k} \right\rfloor\} \\ \mathbf{Y}_j^C &= \{\mathbf{y}_i, i = 1, \ldots, \left\lfloor \frac{n(k-1)}{k} \right\rfloor\}, \qquad \mathbf{Y} = \mathbf{Y}_j \cup \mathbf{Y}_j^C \end{aligned}$$

3. Compute the j-th k-fold error rate e_j^k using only the omitted subset

Repeat these steps k times, error rate is given by

$$e^k = \frac{1}{k} \sum_{j=1}^{k} e_j^k \tag{5.3}$$

10-fold cross-validation displayed lower variance compared to leave-one-out cross-validation [21] because the leave-one-out training sets are too similar to the full data set and so we apply 10-fold cross-validation to our classifiers to asses their performance.

With this procedure, a **confusion matrix** can be constructed. A confusion matrix contains information about actual and predicted classifications done by a classifier. Performance of such systems is commonly evaluated using the data in the matrix. The following table shows the confusion matrix for a two class classifier. a and d is the

true predicted	bad (b)	good (g)
bad (B)	a	b
good (G)	c	d

Table 5.1: Example of a confusion matrix

number of correct predictions of bad and good, respectively, b and c is the number of incorrect predictions of bad and good. In the rest of the chapter, for each classifier, to get reliable estimates we ran 100 cycles of tenfold cross-validation error rate estimation to assess the error rate variation. Thus, the number of correct and incorrect predictions is the average over these cycles, furthermore we normalize the table dividing it by the sum of predictions.

We use boxplots confusion tables and histograms to summarize the error rates. In histograms, the gray vertical line represents CPNS misclassification error (Table 5.3). **k-fold cross-validation** cannot be applied to **PNS error**, indeed the classifier built using a subset in step 2 of Section 5.1 is always the same because it is fixed by the *LABR* ($\mathbf{w} = \mathbf{1}_{p\times 1}$ and $w_0 = 13$). Thus, we cannot obtain its variance.

Bootstrap

Bootstrap is another method for estimating the generalisation error **corrected for bias**. As we saw above, it has proved superior to many other techniques [14]. This technique was first developed by Efron (1979). The basic idea involves sampling with replacement to produce random samples of size m from the original data. Each of these samples is known as a bootstrap sample and each provides and estimate of the bias. Repeating the sampling a large number of times provides information on the variability of the estimator, approximatively 100 subsamples might be used. It is implemented as follows

1. Generate the j-th bootstrap data set $\mathbf{y_j}$ according to a uniform $1/m$ distribution
2. Generate the classifier using $\mathbf{y_j}$
3. Compute the j-th apparent error rate e_j^A using the latter classifier (see Equation 5.2)
4. Calculate the actual error rate e_j^C considering our whole sample $\mathbf{y}$
5. Compute $w_j = e_j^A - e_j^C$

Repeat these steps B times and compute the bootstrap bias

$$W_{boot} = \frac{1}{B}\sum_{j=1}^{B} w_j \tag{5.4}$$

The bias-corrected **apparent error rate** is given by

$$e_{boot}^A = e^A - W_{boot} \tag{5.5}$$

With the basic bootstrap procedure, the apparent error rate e_j^A is obtained by considering the bootstrap data, so the same data as held in the classifier, thus a confusion

matrix will be dramatically biased and is not enlightening. In the following, only the bootstrap error and its deviance is used.

In the rest of the chapters, for each classifier we ran 50 cycles of bootstrap with 100 bootstrap samples each time. Misclassification rate estimation is compute and the 50 cycles allows us to assess the error rate variation.

5.2 Comparison between classification strategies

In order to assess the benefits of splitting the dataset into fresh and cryo zygotes the error obtained with the splitted data set and the error obtained with all data are compared. The error for a given classifier using all data will be referred to as e_{all}, while the error with the splitted dataset is called e_{split}. The whole error rate for splitted data is computed as

$$e_{split} = e_{fresh} p(w_{fresh}) + e_{cryo} p(w_{cryo}) \tag{5.6}$$

where e_{fresh} is the classification error for the fresh data and $p(w_{fresh})$ is the estimates of the a priori probabilities of conservation membership. In our dataset there are as many fresh zygotes as cryo ones, thus

$$p(w_{fresh}) \approx p(w_{cryo}) \approx 0.5$$

5.3 Linear discriminant analysis

assumptions:	multivariate normal class densities with same covariance matrix in each class
space partitioning:	hyperplanes
free parameters:	a priori probability of classes
advantages:	rescale of the data so that the within class covariance is spherical, ease of interpretation and ease of use
disadvantage:	restrictive assumption of class densities, data must be full rank (at least as many observations as features)

This section is dedicated to linear discriminant analysis (LDA), it is a classical statistical approach for predicting samples of unknown classes, based on training samples with known classes. Linear Discriminant Analysis easily handles the case where the within-class frequencies are unequal and their performances has been examined on randomly generated test data. This method maximizes the ratio of between-class variance to the within-class variance in any particular dataset guaranteeing maximal separability. LDA is studied in hopes of providing better classification compared to Principal Components Analysis. The prime difference between LDA and PCA is that PCA does more of feature classification and LDA does data classification. In PCA, the shape and location of the original data sets changes when transformed to a different space whereas LDA does not change the location but only tries to provide more class separability and draw a decision region between the given classes.

5.3.1 Mathematical background

LDA seeks a linear combination $h(\mathbf{x})$ of components of $\mathbf{x} = (x_1, \ldots, x_p)^T$ so that the different classes are separated by hyperplanes (see Equation 5.7), $\mathbf{w}$ is called weight

vector and w_0 the threshold . Function $h(\mathbf{x})$ is the equation of the hyperplane with perpendicular direction $\mathbf{w}$ and a perpendicular distance $|w_0|/|w|$ from the origin [14]. $h(\mathbf{x})/|\mathbf{w}|$ is the perpendicular distance from the pattern $\mathbf{x}$ to the hyperplane.

$$h(\mathbf{x}) = \mathbf{w^T x} + w_0 \tag{5.7}$$

A sample $\mathbf{x}$ is assigned to bad if $h(\mathbf{x}) \geq 0$. Otherwise, it is assigned to good. Let $\mathbf{x}_{jg}$ $(p \times 1)$ be the observation within class g containing p-features. The sample within group g has n_g elements and the mean of $\mathbf{x}_{jg}$ is denoted by $\bar{\mathbf{x}}_g$ and obtained with Equation 5.8, $\bar{\mathbf{x}}$ is the mean of p-features considering both groups.

$$\bar{\mathbf{x}}_g = \frac{1}{n_g} \sum_{i=1}^{n_g} \mathbf{x}_{ig} \tag{5.8}$$

Now let us find the direction $\mathbf{w}$. There is a space of best separation for a determined $\mathbf{w}$ maximizing Equation 5.9.

$$\phi = \frac{\mathbf{w^T B w}}{\mathbf{w^T W w}} \tag{5.9}$$

where $\mathbf{W}$ denotes the within-class covariance matrix (Equation 5.10) and $\mathbf{B}$ the between class covariance matrix (Equation 5.11).

$$\mathbf{W} = \sum_{j=1}^{n_g} \sum_{g=1}^{G} (\mathbf{x}_{jg} - \bar{\mathbf{x}}_g)(\mathbf{x}_{jg} - \bar{\mathbf{x}}_g)^T \tag{5.10}$$

$$\mathbf{B} = \sum_{g=1}^{G} n_g (\bar{\mathbf{x}}_g - \bar{\mathbf{x}})(\bar{\mathbf{x}}_g - \bar{\mathbf{x}})^T \tag{5.11}$$

We choose the direction $\mathbf{w}$ that maximizes $\phi(\mathbf{w})$, differentiating $\phi(\mathbf{w})$ and setting the derivative to zero to find the maximum

$$\frac{d\phi}{dw} = 0 = \frac{2\mathbf{Bw}(\mathbf{wWw}) - 2(\mathbf{wBw})\mathbf{Ww}}{(\mathbf{w^T W w})^2} \tag{5.12a}$$

$$\text{equivalent to} \qquad \mathbf{Bw} = \mathbf{Ww}\left(\frac{\mathbf{wBw}}{\mathbf{wWw}}\right) = \phi\mathbf{Ww} \tag{5.12b}$$

$$\text{iff } \mathbf{W} \text{ not singular} \qquad \mathbf{W^{-1}Bw} = \phi\mathbf{w} \tag{5.12c}$$

In order to be able to compute the inverse of the within class covariance matrix, the matrix must be full rank, this can be only the case if the data set has at least as many rows as columns (necessary condition).

Equation 5.12c implies that $\mathbf{w}$ must be an eigenvector of $\mathbf{W^{-1}B}$ with eigenvalue ϕ. Thus, $\mathbf{w}$ is the eigenvector of higher eigenvalue. In our case $G = 2$, matrix $\mathbf{B}$ of rank one can be written as follows

$$\mathbf{B} = \frac{n_1 n_2}{n_1 + n_2}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_1) \tag{5.13}$$

$\mathbf{W^{-1}}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)$ is the eigenvector associated to the eigenvalue of the matrix $\mathbf{W^{-1}B}$, so

$$\mathbf{w} \propto \mathbf{W^{-1}}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2) \tag{5.14}$$

We normalize $\mathbf{w}$ without loss of generality, the weight vector becomes

$$\mathbf{w} = \frac{\mathbf{W^{-1}}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)}{\|\mathbf{W^{-1}}(\bar{\mathbf{x}}_1 - \bar{\mathbf{x}}_2)\|} \tag{5.15}$$

The classification rule for a new observation is to assign $\mathbf{x}$ to bad if $h(\mathbf{x}) = \mathbf{w^T x} + w_0$ is closer to $\bar{y}_1 = \mathbf{w^T \bar{x}_1}$ than to $\bar{y}_2$, or

$$h(\mathbf{x}) = \mathbf{w^T x} - \frac{1}{2}(\bar{\mathbf{x}}_\mathbf{1} + \bar{\mathbf{x}}_\mathbf{2}) > 0 \tag{5.16}$$

Another approach is to assume that the class-wise densities are multivariate Gaussian distributions with means $\bar{x}_1$ and $\bar{x}_2$, and common covariance matrix C (see Equation 4.13a). Then, the normal distribution for both classes becomes

$$p(\mathbf{x}|g) = \frac{1}{\sqrt{(2\pi)^p |C|}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{1})C^{-1}(\mathbf{x}-\bar{\mathbf{x}}_\mathbf{2})} \tag{5.17}$$

Classification is achieved by calculating the posterior probabilities $p(w_i|\mathbf{x})$, and assigning the label to a class for which they are the greatest. Bayes's rule allows us to to compute them

$$p(w_i|\mathbf{x}) = \frac{p(\mathbf{x}|w_i)p(w_i)}{p(\mathbf{x})} \tag{5.18a}$$

or equivalently, (5.18b)

(5.18c)

$$log(p(w_i|\mathbf{x})) = log(p(\mathbf{x}|w_i)) + log(p(w_i)) - log(p(\mathbf{x})) \tag{5.18d}$$

$$= -\frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}}_\mathbf{i})^T C^{-1}(\mathbf{x} - \bar{\mathbf{x}}_\mathbf{i}) - log(p(w_i)) + const \tag{5.18e}$$

$$= \mathbf{x}^T C^{-1}\bar{\mathbf{x}}_i - \frac{1}{2}\bar{\mathbf{x}}_i^T C^{-1}\bar{\mathbf{x}}_i + log(p(w_i)) + const = g_i(\mathbf{x}) \tag{5.18f}$$

(5.18g)

The discriminant rule becomes, assign $\mathbf{x}$ to w_i if $g_i > g_j$, for all $i \neq j$. The means and the covariance matrix are unknown but using the plug-in approach all we have to do is replacing them by their maximum likelihood estimator.

5.3.2 Results

Unweighted LDA with PNS data (CPNS)

As discussed in Section 2.1.1 **CPNS** is the cumulated pronuclear score, this score can be seen as an **LDA** classification with an unitary weight vector $\mathbf{w}$, the threshold w_0 is already established by the laboratory to $w_0 = 15$. This threshold can be obtained form Figure 5.1. The implantation rate is given by $d/(a+d)$ in Table 5.1, this rate increases for all zygotes when the cumulated pronuclear score is greater than 8. At level 15 the implantation rate is maximum. Hence, $w_0 = 15$, Equation 5.7 becomes

$$h(\mathbf{x}) = \sum_{i=1}^{p} x_i + 15 \tag{5.19}$$

The classification becomes if $h(\mathbf{x_i}) \geq 15$, the x_i zygote will be considered as having maximum implantation potential. Right side of Figure 5.1 is the frequency of the CPNS score, zygotes having a score of 12 are 22%, when zygotes having a score of 18 are only 0.5% (only one zygote). Implantation rate for frozen-thawed zygotes is equal to 1 for a CPNS score greater or equal than 15. However, we can not conclude that all cryo-preserved zygotes with a score higher than 15 lead to a pregnancy because only one frozen-thawed has a score over 15, thus the implantation rate becomes $d/(a+d) = 1/(0+1) = 1$.

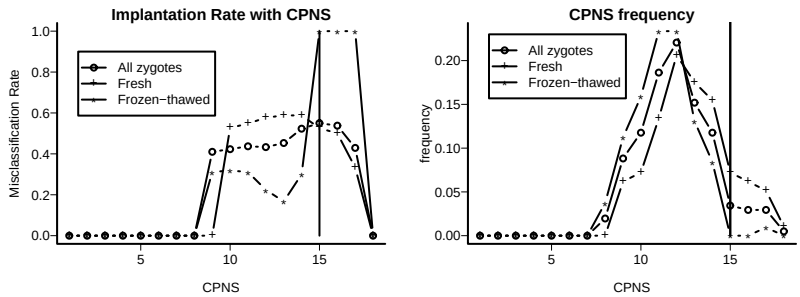


Figure 5.1: Implantation rate as function of the cumulated pronuclear score (CPNS) and CPNS frequency

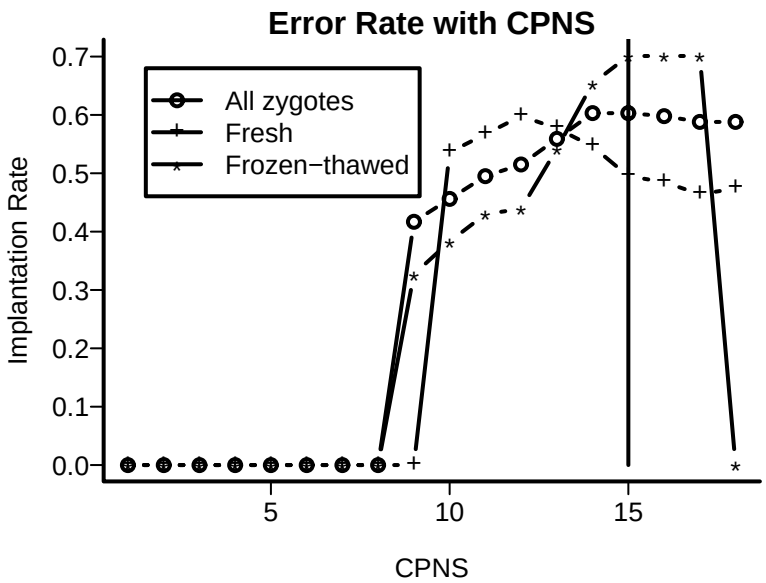


Figure 5.2: Implantation rate as function of the cumulated pronuclear score (CPNS)

Throughout this chapter, **CPNS misclassification error** is compared to the misclassification rate of supervised classifiers. Its confusion matrix and its values are shown in Tables 5.2 and 5.3.

For frozen-thawed zygotes the accuracy of the classifier is very low, in fact neither bad frozen zygote are classified as being bad nor as being good. However, the overall classifier accuracy is of 0.3 (see Table 5.2), this fact can be explained by the small

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.549	0.044	0.392	0.0928	0.692	0.0
good	0.353	0.054	0.412	0.103	0.299	0.009

Table 5.2: Confusion matrix with CPNS procedure (unweighted LDA with PNS data)

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
CPNS error	0.398	0.505	0.3	0.402

Table 5.3: Misclassification error rate with CPNS procedure (unweighted LDA with PNS data)
s

number of thawed-zygotes in our sample, only 26 of 59 ($26/(26+59) \approx 0.3$) are good. Even if we compare all our classifiers with accuracy in Table 5.2 this fact must be taken in account.

Conventional LDA with PNS and IJPM data

Although the variance of the error estimates for the **PNS** data is smaller that those based on the **IJPM** data due to noise (**PNS** can only take 3 values (see Section 2.1.1), whereas values of **IJPM** are continuous), the whole **IJPM** dataset performs better classification than **PNS** in respect to the estimation of the error rate. The mean error estimation for **IJPM** is 0.36 ± 0.02 whereas the mean for **PNS** is 0.39 ± 0.01 (see Table 5.6). Concerning the sampled data (fresh or cryo) the error estimation is lower for the **PNS** dataset. However, if we want to assess accuracy of classification based on error rate estimation procedures, the classification error with divided data e_{split} (cryo or fresh) and with whole data e_{all} must be compared (see Equation 5.6). Within the splitted data the PNS is better than IJPM in respect to misclassification error rate, concerning all data together IJPM is better, hence we compare e_{split}^{PNS} with e_{all}^{IJPM}, finally $e_{all}^{IJPM} < e_{split}^{PNS}$ ($0.363 < 0.396$) (see Tables 5.6). The smallest misclassification error with LDA is obtained with the IJPM dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.363 \pm 0.02}$.

Histograms of classification error rate are shown in Figure 5.4. Vertical gray line represents the error rate with the CPNS scoring in Table 5.3, the black line represents the mean of the error rate with the LDA procedure in Table 5.6. Error rate with LDA procedure (black) is always below the error rate with CPNS scoring (grey), except in the case of thawed-zygotes, however we must keep in mind the fact that almost all cryo-preserved zygotes are misclassified. CPNS error in first histogram (grey line) overlaps the LDA classification with PNS data, that can be also a reason to choose the value 15 for the threshold w_0.

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.385 ± 0.016	0.197 ± 0.016	0.24 ± 0.0231	0.232 ± 0.0231	0.449 ± 0.031	0.245 ± 0.031
good	0.169 ± 0.0142	0.25 ± 0.0142	0.261 ± 0.0209	0.268 ± 0.0209	0.164 ± 0.0148	0.142 ± 0.0148

Table 5.4: Confusion matrix for 10-fold cross-validation for IJPM data

Results with bootstrap confirm those obtained with 10-fold cross-validation, however with this approach has less variance.

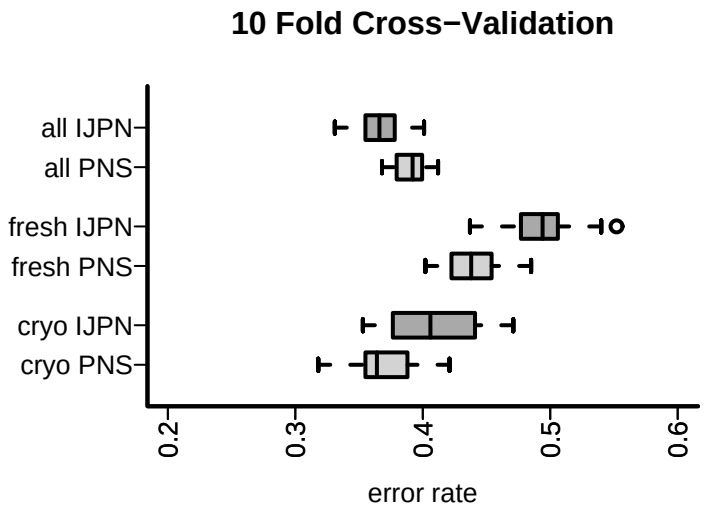


Figure 5.3: 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

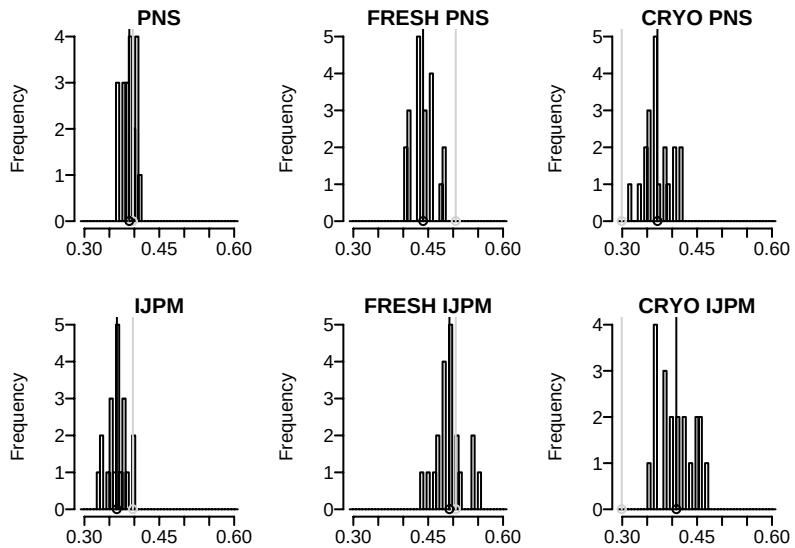


Figure 5.4: Histogram of 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

5.3.3 Conclusion

The main conclusion of this section is that the smallest misclassification error with LDA is obtained with the **IJPM dataset** without distinction between their conservation

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.41 ± 0.00799	0.184 ± 0.00799	0.268 ± 0.014	0.217 ± 0.014	0.481 ± 0.0236	0.21 ± 0.0236
good	0.206 ± 0.00848	0.2 ± 0.00849	0.223 ± 0.0147	0.293 ± 0.0147	0.16 ± 0.00798	0.148 ± 0.00797

Table 5.5: Confusion matrix for 10-fold cross-validation for PNS data

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.393 ± 0.0132	0.446 ± 0.0245	0.347 ± 0.0189	0.396 ± 0.0154
IJPM	0.363 ± 0.0214	0.497 ± 0.0351	0.391 ± 0.0304	0.444 ± 0.0232

Table 5.6: Summary of 10-fold error rates

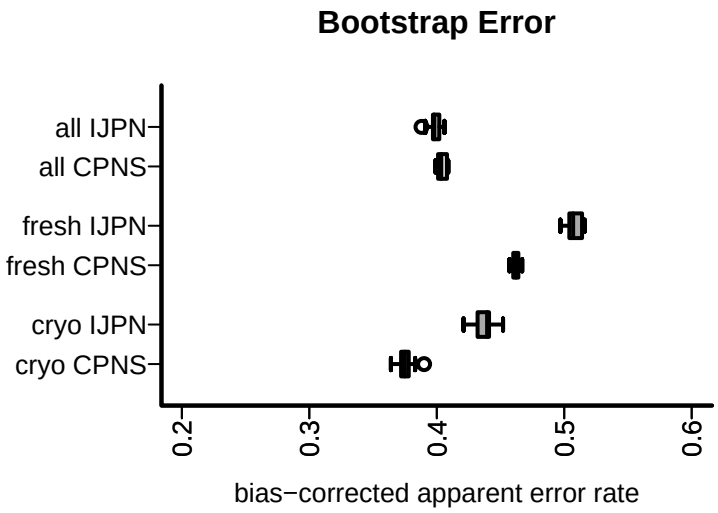


Figure 5.5: Bootstrap error rate both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.404 ± 0.00386	0.462 ± 0.00307	0.376 ± 0.00726	0.405 ± 0.0184
IJPM	0.398 ± 0.00549	0.508 ± 0.00606	0.439 ± 0.00929	0.378 ± 0.00942

Table 5.7: Summary of bootstrap error rates

type, its value is $\mathbf{0.363 \pm 0.02}$ for 10-fold cross-validation.

5.4 Trees

assumptions:	hyperspace separation with stepwise method
space partitioning:	smaller hypercubes within hypercubes
free parameters:	α amount of penalisation, $I(t)$ impurity function, f number of features per node
advantages:	ease of interpretation and ease of use
disadvantage:	stepwise method, high variance and unstable classifiers

This section is dedicated to classification trees. Trees are usual classifiers in medical diagnosis for their ease of interpretation and their compactness. Trees have demonstrated good performance on a wide variety of problems in which there is not a strong theory in an area that would clearly indicate which variables are, and are not, probably predictors of some dependent classification. Trees will be very useful in identifying major data trends, and hence this method is well adapted for our fixed goals. Each endpoint of our classification tree allows us to predict whether an expected observation belongs to the "good" or the "bad" class zygotes leading to pregnancy or zygotes not leading to pregnancy respectively.

5.4.1 Mathematical background

Tree models used throughout this section, are hierarchical classifiers based on binary logical rules. They divide the space into **multidimensional rectangles**. In fact, a tree cuts the feature space into smaller and smaller hypercubes, each hypercube corresponds to a terminal node of the tree. Trees are usually displayed in a graph which has the format of a binary decision tree, each tree has grows from the root node to the terminal nodes (also called leaves). Three steps in the algorithm to find the "best" tree are

1. search the j-th feature that should be used at the i-th node to split the samples into subgroups
2. select the threshold x_{th} on j-th feature that should be assigned at the i-th node (an example of decision is $\{\mathbf{x} \in \Re : x_j \leq x_{th}\}$)
3. choose which leaf is a terminal node

We first describe **splitting rules**, including the first two steps, with our features based on the *Gini criterion*. Then, stopping rules to assign terminal nodes.

Splitting and stopping rules

Following L. Breiman (1984) [22], the splits are carried out in a way that the selected j-th feature and threshold at a certain node get as "pure" as possible. The decrease in impurity at each node t is measured with an **impurity function** $I(t)$. A common impurity function is defined by the *Gini criterion* (5.20a).

$$I(t) = \sum_{i \neq j} p(w_i|t)p(w_j|t) \tag{5.20a}$$

$$p(w_i|t) = N_j(t)/N(t) \tag{5.20b}$$

where $I(t)$ is the impurity at the node t, $p(w_i|t)$ is an estimate of the probability distribution over w_i, $N_j(t)$ is the number of samples of class w_j leading to the node t and $N(t)$ is the total number of samples in that node ($N(t) = \sum_j^2 N_j(t)$).

The impurity function $I(t)$ at node t allows us to choose the split that minimizes the impurity at the two leafs leading to nodes t_L and t_R left node and right node respectively, the decrease in impurity function is

$$\delta I(t) \equiv \sum_{i \neq j} p(w_i|t)p(w_j|t) - (I(t_R)p_R + I(t_L)p_L) \tag{5.21a}$$

$$\equiv \sum_{i \neq j} p(w_i|t)p(w_j|t) - (\sum_{i \neq j} p(w_i|t_R)p(w_j|t_R)p_R + \sum_{i \neq j} p(w_i|t_L)p(w_j|t_L))p_L) \tag{5.21b}$$

Now we can grow our tree one step lookahead ("stepwise" method) selecting nodes maximizing the decrease in impurity over our features. The process described above can be applied until each terminal node contains only one observation, but often leads to an **over-fit** to the training data and the classification could not prove useful. We, thus, must stop the splitting rule process or **prune** the tree having "pure" (all samples that arrive at that node belong to the same class) or nearly pure class membership. In the following section we will describe a standard pruning algorithm to determine which leaf is a terminal node.

Pruning the tree

The most popular pruning approach is the one proposed by Breiman (1984) [22]. The basic idea this approach is that too big trees (having numerous leaves) yield an over-fit. On the other hand, the misclassification rate $R(t)$ will increase if the number of leaves is too little. So, a compromise is searched with the cost-complexity measure.

Firstly, let's make some definitions. T_i is a pruned subtree of T if it has tthe same root node. Let $\tilde{T}_i$ denote the set of terminal nodes of the tree T_i, then let $N(\tilde{T}_i)$ denote its cardinality. Finally, let α be the amount of cost of complexity (control of penalization) of $N(\tilde{T}_i)$. The penalized misclassification rate for the node t is $R_\alpha(t)$ and can be computed as follows

$$R_\alpha(t) = R(t) + \alpha \tag{5.22a}$$

$$R_\alpha(T_i) = \sum_{t \in \tilde{T}_i} (R(t) + \alpha) = R(T_i) + \alpha N(\tilde{T}_i) \tag{5.22b}$$

When $R_\alpha(T_i) = R_\alpha(t)$ the contribution to the cost of complexity of the subtree T_i is the same than that for the node t and α can be write as a function of node t

$$\alpha(t) = \frac{R(t) - R(T_i)}{N(\tilde{T}_i) - 1} \tag{5.23a}$$

The numerator is the difference between the estimated misclassification error for the node t and the estimated misclassification error for the whole subtree with the same root, the denominator is the number of terminal nodes of the subtree. If $\alpha = 0$, there will be no pruning (no penalization). On the other extreme, if we chose $\alpha = +\infty$ all nodes except the root node are removed. In practice, we can compute $R(T_i)$ by

estimating the amount of misclassification error at node t, $R(T_i)$ can also be estimated. It is the amount of misclassification error for the subtree T_i, and $N(\tilde{T}_i)$ the cardinality of T_i. The only free parameter is the cost-complexity $\alpha(t)$ reflecting the compromise described above. The following process is used to prune the tree T

1. search a node t_k (can be more than one) with the smallest value of $\alpha(t) = \alpha_k$
2. make it a terminal node, the new tree without the sub-branches from t_k node is labeled T^k
3. calculate the new $\alpha(t)$ for its ancestors, always superior to α_k

This process is iterated k times until the terminal node is the root node. Now we have a collection of trees T^k ($k \in \{1,\ldots,k\}$) and corresponding α_k.

Now, we can grow an the prune our tree that fits our training set, but it can be inefficient in classification specially when the available dataset contains less than a thousand cases [22], thus cross-validation is used to select the best tree, enhancing the predictability of our classifier.

Selecting the optimal tree

After the pruning of tree T iterated k times, we have a collection of trees T^k ($k \in \{1,\ldots,k\}$), 10-fold cross-validation is applied, we take 90% of the sample, grow the tree using this part of the sample, prune a sequence of subtrees and calculate the error rate for every subtree in the sequence using the rest 10% of the sample as a test set. This is repeated 10 times, every time using different part of the sample as an estimation set and as a test set. The problem is that we have different data to grow and to prune, thus, α_k sequences are differents. The approach described by Breiman ([22]) is to first grow and prune using all of he data, whichs gives a sequence α_k, then form a new sequence $\hat{\alpha}_k = \sqrt{\alpha_k \alpha_{k+1}}$ (geometric mean). When pruning trees grown with 90% of the sample, we choose the best tree from the sequence which minimize the misclassification estimate $R^{CV}(T^k)$.

Let $T(\alpha)$ the pruned subree with all nodes having $g(t) > \alpha$, ($T(\alpha) = T^k$ at the k-th stage). 10-fold cross-validation is used to prune the tree the method is as follows

1. use the prune process to generate a sequence of pruned subtrees T^k of T
2. generate V subsets $L_v \in \{L_1,\ldots,L_V\}$ into V parts with approximately equal size and generate a tree T_v
3. use the prune process to generate a sequence of pruned subtrees T^{vk} of T_v (5.4.1)
4. compute the cross-validation estimate of misclassification $R^{CV}(T^k)$

$$R^{CV}(T^k) = \frac{1}{V}\sum_{v=1}^{V} R_v(T_v(\hat{\alpha}_k)) \tag{5.24}$$

   where, R_v is the estimate of misclassification based on the subset L_v for the tree $T_v(\hat{\alpha}_k))$ and $\hat{\alpha}_k$ is the geometric mean.

5. select the smallest tree $T_{min} \in \{T^1,\ldots,T^K\}$

$$R^{CV}(T_{min}) = \min_k R^{CV}(T^k) \tag{5.25}$$

The final pruned tree is T_{min}.

5.4.2 Results

For the tree approach, the variance of the error estimates for the **PNS** data is also is smaller that those based on the IJPM data certainly due to noise as seen for LDA. The whole PNS dataset performs better classification than IJPM in respect to the estimation of the error rate. As the quartiles in 10-fold cross-validation overlap, the difference is not significant. Concerning the sampled data (fresh or cryo) the error estimation is lower for PNS dataset.

In order to assess the benefits of splitting the dataset into fresh and cryo zygotes, the error obtained with the splitted data set and the error obtained with all data are compared. Within the splitted data the PNS is better than IJPM in respect to misclassification error rate (0.423 < 0.469), concerning all data together PNS is also better (0.423 < 0.414), results are showed in Table 5.10). Hence, we compare e^{PNS}_{split} with e^{PNS}_{all}, finally $e^{PNS}_{all} < e^{PNS}_{split}$ (0.414 < 0.423). The smallest misclassification error with trees is obtained with the PNS dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.41 \pm 0.02}$.

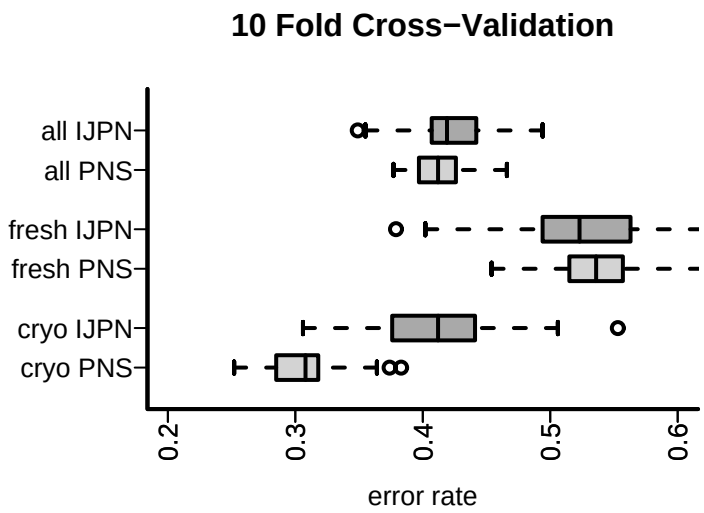


Figure 5.6: 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

Histograms of classification error rate are shown in Figure 5.7. Vertical gray line represents the error rate with the CPNS scoring in Table 5.3, the black line represents the mean of the error rate with tree procedure in Table 5.6. Error rate with trees procedure (black) always overlap the error rate with CPNS scoring (grey), except in the case of thawed-zygotes. So **classification with trees using IJPM dataset do not improve the classification**. However, we must keep in mind the fact that almost all cryopreserved zygotes are misclassified. CPNS error in first histogram (grey line) overlaps the tree classification with PNS data, as seen for LDA.

As the whole PNS dataset without splitting the dataset performs slightly better classification than IJPM in respect to the estimation of the error. We grow as an example, a randomly chosen tree among the 100 trees with PNS dataset (Figure 7.3). In our trees,

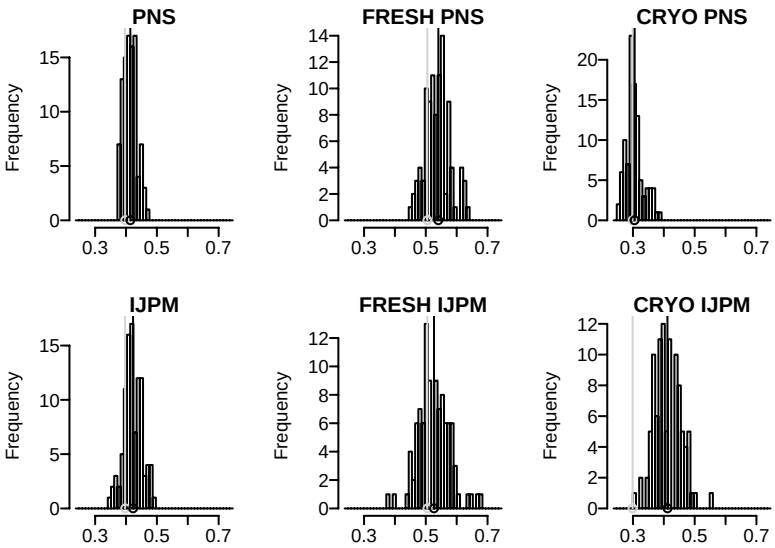


Figure 5.7: Histogram of 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

categorical features are treated as numerical data, for example ICSI can take categorical values "yes" or "no", respectively for Intra-Cytoplasmic Sperm Injection ICSI and in vitro fertilisations, these values are converted to numerical values 1 and 0. Thus, the threshold in node ICSI of Figure 7.3 split the ICSI fertilisation to the left if it is greater or equal to 0.5 ("yes" label) and in vitro to the right ("no" label). Values at each terminal node are the class counts, for example for PNS cryo data, the first terminal node has a value of 62/18 which means that 62 bad zygotes and 18 good zygotes fall in this terminal node. Ideally, we would like 100% of classification 70/0 in this latter node. It is quite possible to grow the tree that fits the training so well, but it leads to a too well adaptation to the features of that subset resulting in over-fit.

Although the misclassification rate with the uniweighted LDA (CPNS) is lower than with the trees (0.398 < 0.41), the tree in Figure 7.3 can be useful to the *LABR* for his ease of interpretation and ease of use. Furthermore, many zygotes have not a CPNS score superior to 15, indeed the probability in our data set having a zygote with a CPNS superior to 15 was of 10% (see Figure 5.2). And so, the advantage of the trees is that they can be used even if the CPNS score is low.

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.362 ± 0.0288	0.22 ± 0.0288	0.216 ± 0.0355	0.255 ± 0.0355	0.485 ± 0.041	0.209 ± 0.041
good	0.203 ± 0.0214	0.215 ± 0.0214	0.271 ± 0.0404	0.257 ± 0.0404	0.202 ± 0.0244	0.103 ± 0.0244

Table 5.8: Confusion matrix for 10-fold cross-validation for IJPM data

5.4.3 Conclusion

Trees approach lead to an overall higher misclassification rate compared to LDA, furthermore trees show lower stability, interquartile ranges are broader than those previ-

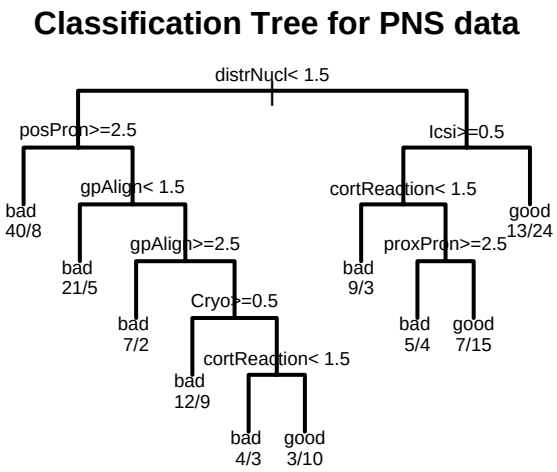


Figure 5.8: Classification Tree for PNS data

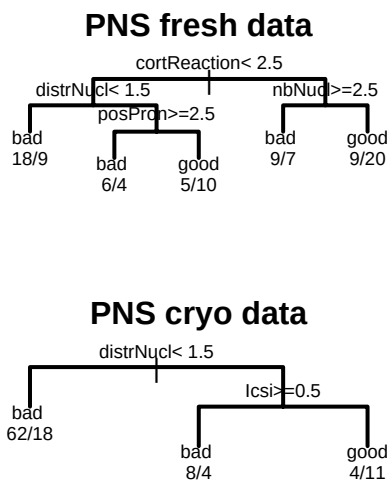


Figure 5.9: Classification Tree for PNS divided data

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.39 ± 0.0165	0.203 ± 0.0165	0.2 ± 0.0312	0.284 ± 0.0312	0.586 ± 0.0292	0.105 ± 0.0292
good	0.211 ± 0.0152	0.196 ± 0.0152	0.256 ± 0.0323	0.259 ± 0.0323	0.2 ± 0.0084	0.108 ± 0.00839

Table 5.9: Confusion matrix for 10-fold cross-validation for PNS data

ously computed with LDA. Therefore, our preliminary analysis suggest that trees in

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.414 ± 0.0207	0.541 ± 0.0402	0.306 ± 0.0292	0.423 ± 0.024
IJPM	0.423 ± 0.0296	0.527 ± 0.0515	0.412 ± 0.044	0.469 ± 0.0355

Table 5.10: Summary of 10-fold error rates

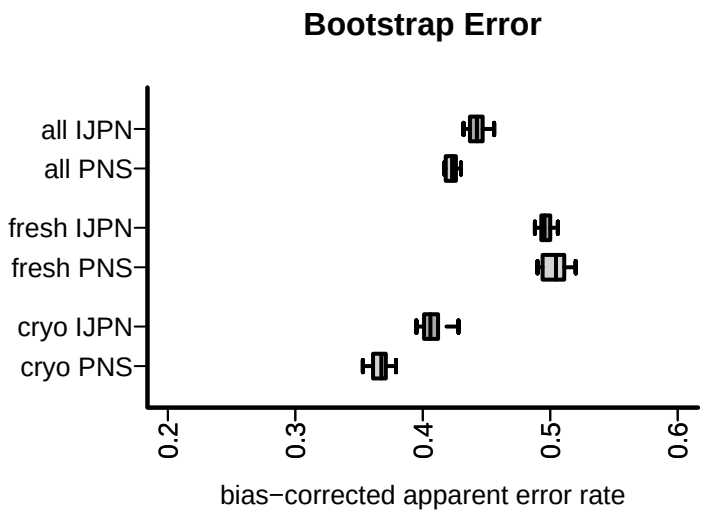


Figure 5.10: Bootstrap error rate both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.423 ± 0.00432	0.503 ± 0.00973	0.367 ± 0.00802	0.435 ± 0.00725
IJPM	0.443 ± 0.00797	0.496 ± 0.00519	0.407 ± 0.00951	0.452 ± 0.00486

Table 5.11: Summary bootstrap error rates

our case are less useful than LDA. However, trees are a useful method to summarize data and to make a quick human diagnostic. Indeed, two quick human diagnostics can be made before the transfer, the first is to calculate the CPNS score and the second is to drop the zygote down the tree of Figure 7.3. The advantage of using our tree classifier is that a zygote with a score under 15 is considered as any zygote. Hence, allowing a classification with all zygotes as opposed to the CPNS classification. This advantage is that in our sample, only 10% zygotes have a score over or equal to 15 (see Figure 5.2).

The main conclusion of this this section is that the smallest misclassification error with tree is obtained with the PNS dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.41 \pm 0.02}$ for 10-fold cross-validation.

5.4.4 Mathematical background

5.5 Support Vector Machines

assumptions:	the larger the margin, the better the generalisation error of the classifier
space partitioning:	non-linear frontiers
free parameters :	choice and parameters of Kernel K, regularisation parameter C
advantages:	more powerful than LDA, not a stepwise method
disadvantage:	not scale-invariant

Support vector machine (SVM) model is a nonlinear discriminant analysis. The basic idea of SVM is to seek for a separating non-linear decision boundary (see Equation 5.26) maximizing a margin ρ between two classes y_1 and y_2 with a higher dimensional feature space. This can be achieved if the data is separable, but in practice the data is not separable, in that case a variable must be introduced to allow the constraints to be met.

5.5.1 Mathematical background

Separable data

For a nonlinear function ϕ, **SVM** seeks a non-linear decision boundary which maximizes the margin ρ, the assumption is that the larger the margin, the better the generalisation error of the classifier, ω is called weight vector and ω_0 the bias. Classes w_1 and w_2 have corresponding labels $y_1 = 1$ and $y_2 = -1$. Let n be the number of observations, the margin satisfies for both classes

$$y_j \left(\sum_{i=1}^{m} \omega_i \phi_i(\mathbf{x_j}) + \omega_0 \right) = y_j \left(\omega^{\mathbf{T}} \phi(\mathbf{x_j}) + \omega_0 \right) \geq \rho \tag{5.26}$$

Equation 5.26 is the equation of two parallel non-linear boundaries on each side of $\omega^{\mathbf{T}}\phi(\mathbf{x_j}) + \omega_0 = \rho$ separated by the distance $\rho / \|\omega\|$ (hyperplanes). Equivalently, we can divide ρ and ω_0 in Equation 5.26 by ρ, the new boundaries are called **canonical decision boundaries**

$$y_j \left(\omega^{\mathbf{T}} \phi(\mathbf{x_j}) + w_0 \right) \geq 1 \tag{5.27a}$$

$$\text{and let,} \quad f(\mathbf{x_j}, y_j) = y_j \left(\omega^{\mathbf{T}} \phi(\mathbf{x_j}) + w_0 - 1 \right) \tag{5.27b}$$

and so the distance becomes $1/\|\omega\|$. The points that lie on the canonical boundaries are called **support vectors**. The maximization of the distance becomes the minimization of $\|\omega\|$. The problem is minimizing $\|\omega\|$ subject to constraints (5.27b), the corresponding *Lagrangian* is

$$L = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{j=1}^{n} \lambda_j f(\mathbf{x_j}, y_j) \tag{5.28}$$

where $\{\lambda_j, i = 1, \ldots n; \lambda_j \geq 0\}$ are the *Lagrangian* multipliers. To minimize Equation 5.28 is equivalent to maximize the distance between two SVM points $(1/\|\omega\|)$ and to minimize the distance of the same points to the boundaries of SVM.

The dual *Lagrangian* problem is to maximize

$$D = \sum_{j=1}^{n} \lambda_j - \frac{1}{2} \sum_{j=1}^{n} \sum_{k=1}^{n} \lambda_j \lambda_k y_j y_k \phi(\mathbf{x_j})^T \phi(\mathbf{x_k}) \tag{5.29}$$

over all $\lambda_j \geq 0$ with $\sum_{j=1}^{n} \lambda_j y_j = 0$. The weight vector $\mathbf{w}$ can be obtained via

$$\mathbf{w} = \sum_{j=1}^{n} \lambda_j y_j \phi(\mathbf{x_j}) \tag{5.30}$$

and w_0 can be computed using any set of support vectors x_j^{SV} with associated values of $\lambda_j^{SV} > 0$ with Equation 5.27b.

Non separable data

For non separable data, the margin in Equation 5.27a must be adjusted by a slack variable $\sigma_j \geq 0$, Equation 5.27a and Equation 5.28 become

$$y_j \left(\omega^{\mathbf{T}} \phi(\mathbf{x_j}) + w_0 \right) \geq 1 - \sigma_j \tag{5.31a}$$

$$L = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{j=1}^{n} \sigma_j - \sum_{j=1}^{n} \lambda_j f(\mathbf{x_j}, y_j) \tag{5.31b}$$

the corresponding dual problem is maximized over all $0 \leq \lambda_j \leq C$ with $\sum_{j=1}^{n} \lambda_j y_j = 0$. To minimize Equation 5.31b is equivalent to maximize the distance between two SVM points $(1/\|\omega\|)$, to minimize the distance of the same points to the boundaries of SVM and the cost for those points that are not inside of the boundaries.

The weight vector $\mathbf{w}$ can be obtained via

$$\mathbf{w} = \sum_{j=1}^{n} \lambda_j y_j \phi(\mathbf{x_j}) \tag{5.32}$$

and w_0 can be computed using any set of support vectors x_j^{SV} with associated values of $0 \leq \lambda_j^{SV} \leq 0$ with Equation 5.31b.

The prediction formula for an observation in SVM $\{x_j, j \in SV\}$ (set of support vectors with associated values of λ_j) is

$$g(\mathbf{x}) = sign\left(\omega^T \phi(\mathbf{x}) + w_0\right) = sign\left(\sum_{j \in SV}^{n} \lambda_j y_j \phi(\mathbf{x_j})^T \phi(\mathbf{x}) + w_0 \right) \tag{5.33}$$

$\phi(\mathbf{x})^T \phi(\mathbf{y})$ is also called **Kernel function** $K(\mathbf{x}, \mathbf{y})$, Equation 5.33 can be replaced by

$$g(\mathbf{x}) = sign\left(\sum_{j \in SV}^{n} \lambda_j y_j K(\mathbf{x_j}, \mathbf{x}) + w_0 \right) \tag{5.34}$$

5.5.2 Results

As we have presented kernels, they can sometimes improve the separability of the two classes of samples by performing an projection of the data into a higher dimensional feature space. Choice of the kernel the actual form of the nonlinearity is relatively

unimportant [14], we choose the polynomial basis function kernels. However, degrees of freedom (dof) and costs (C) may improve the classification, they can be obtained by an iterative process choosing dof and C for which misclassification error rate is a minimum. The process is carried out with the data, results are shown in Table 5.12. These results are used to design our SVM classifiers.

	PNS	PNS fresh	PNS cryo	IJPM	IJPM fresh	IJPM cryo
dof	3	5	4	2	4	5
C	3	1	3	9	1	3

Table 5.12: Minimum misclassification error rate according to the number of degrees of freedom and the costs

Once again, the whole **IJPM** dataset performs better classification than **PNS** in respect to the estimation of the error rate. The mean error estimation for **IJPM** is 0.378 ± 0.02 whereas the mean for **PNS** is 0.392 ± 0.02 (see Table 5.15). Concerning the sampled data (fresh or cryo) the error estimation is also lower for the **IJPM** dataset. Within the divided data the IJPM performs slightly better classification than PNS in respect to misclassification error rate (see Tables 5.15). The smallest misclassification error with SVM is obtained with the IJPM dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.378 \pm 0.02}$.

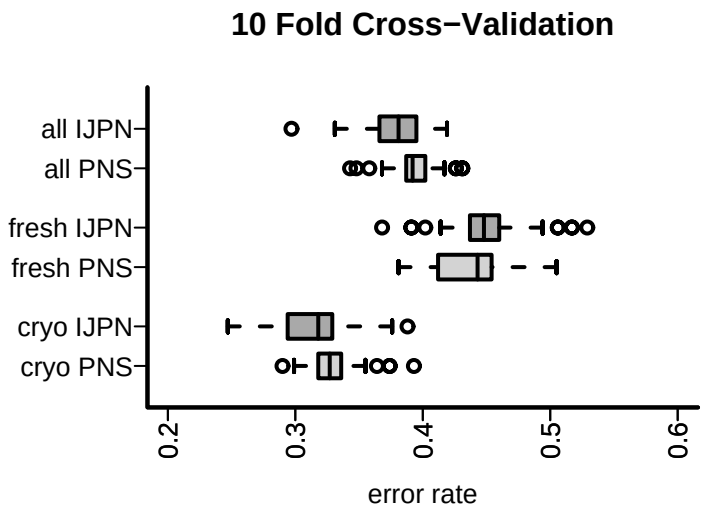


Figure 5.11: 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

Histograms of classification error rate are shown in Figure 5.12. Vertical gray line represents the error rate with the CPNS scoring in Table 5.3, the black line represents the mean of the error rate with SVM procedure in Table 5.15. Error rate with SVM procedure (black) is always below the error rate with CPNS scoring (grey), except in the case of thawed-zygotes, this is the same result as for LDA and trees classifiers. Once again, we must keep in mind the fact that almost all cryo-preserved zygotes are

misclassified.

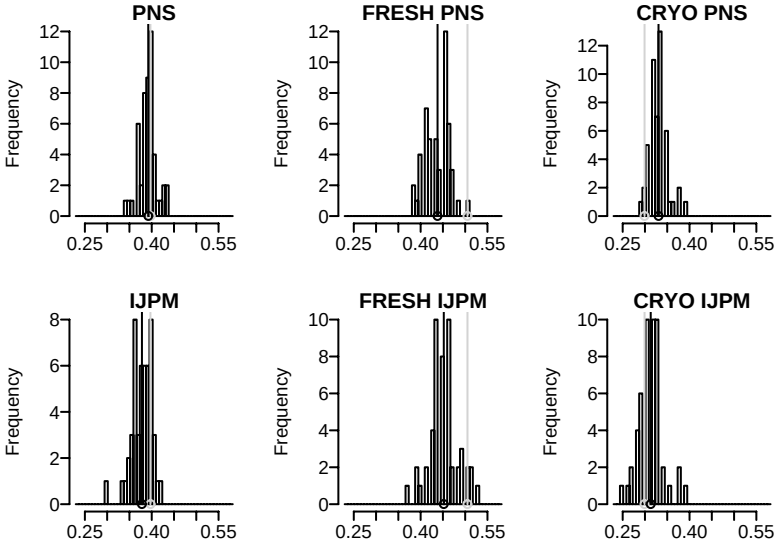


Figure 5.12: Histogram of 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.418 ± 0.0156	0.163 ± 0.0156	0.225 ± 0.0222	0.246 ± 0.0222	0.601 ± 0.0188	0.0927 ± 0.0188
good	0.215 ± 0.0156	0.204 ± 0.0156	0.206 ± 0.0209	0.323 ± 0.0208	0.22 ± 0.0171	0.0856 ± 0.0171

Table 5.13: Confusion matrix for 10-fold cross-validation for IJPM data

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
bad	0.425 ± 0.0143	0.169 ± 0.0143	0.225 ± 0.017	0.26 ± 0.017	0.59 ± 0.0169	0.102 ± 0.0169
good	0.224 ± 0.0118	0.183 ± 0.0118	0.178 ± 0.0209	0.337 ± 0.0209	0.229 ± 0.00841	0.0796 ± 0.00841

Table 5.14: Confusion matrix for 10-fold cross-validation for PNS data

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.392 ± 0.0185	0.438 ± 0.0277	0.33 ± 0.0198	0.384 ± 0.0176
IJPM	0.378 ± 0.0226	0.452 ± 0.0332	0.313 ± 0.0274	0.382 ± 0.0202

Table 5.15: Summary of 10-fold error rates

5.5.3 Conclusion

SVM approach lead to an overall lower misclassification rate compared to LDA, however, they show lower stability. The main conclusion of this this section is that the

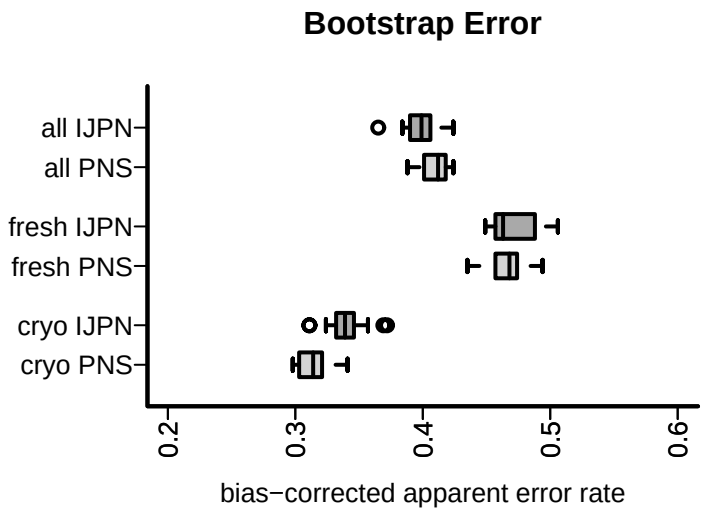


Figure 5.13: Bootstrap error rate both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.41 ± 0.0102	0.467 ± 0.0158	0.314 ± 0.0116	0.39 ± 0.00822
IJPM	0.399 ± 0.0135	0.471 ± 0.0178	0.34 ± 0.0148	0.405 ± 0.011

Table 5.16: Summary bootstrap error rates

smallest misclassification error with SVM is obtained with the IJPM dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.378} \pm \mathbf{0.02}$ for 10-fold cross-validation. The misclassification error obtained with the PNS dataset, for 10-fold cross-validation is of 0.384 ± 0.0176, latter we will see that SVM leads to the best classification for the PNS dataset.

5.6 Random Forest

assumptions:	none
space partitioning:	hypercubes "bagging"
free parameters:	number of variables in the random subset at each node, number of trees in the forest
advantages:	more robust against overfiting, more robust than other classifiers with respect to noise
disadvantage:	"black box" approach

A **Random Forest** is a classifier that is built from multiple trees generated from random sampling of the observations, and the features. One of the reason for performing random forest to our data is that random forest is robust against over-fitting [23],

normally leading to poor performance. Another reason for choosing Random Forest is that Random Forest is more robust with respect to noise than other classification methods [23]. And as seen with PCA, noisy IJPM features has a strong contribution to the overall variance for the IJPM dataset.

Ideally, we would want to find a biological meaning of a correctly classified sample, hence understanding Random Forest "black box" is needed. For this purpose, internals estimates of variable importance are computed in Section 5.6.4

5.6.1 Mathematical background

As seen in Section 5.4, each node in standard trees are split using the best split among all variables, whereas in a random forest, each node is split using the best among a subset of features randomly chosen at that node. The random forest process involves four steps. Let n be the number of samples, and let L ($n \times p$) be the whole training set.

1. Sample n bootstrap subsets $L = \{L_1, \ldots, L_n\}$ with replacement
2. Use L_v to construct the tree classifier T_v without pruning (see Section 5.4), at each node of T_v, m split random features are selected, the best split from these features is retained.
3. T_v are used to predict samples that are not in L_v, called *out-of-bag* and labeled L_v^C ($L = L_v \cup L_v^C$)
4. Final prediction is unweighted prediction for each *out-of-bag* estimators

A **variable importance estimation** can also be computed before using T_v to predict the out-of-bag data (Step 3). If we randomly permute the value for one variable for the out-of-bag samples, this will result in substantially decreasing our ability to classify each individual in the sample, we can find a variable importance estimation.

The size of each bootstrap sample is equal to the size of the original training set, but they are drawn with replacement, so each one contains some duplicates of certain training points and leaves out other training points completely.

5.6.2 Results

Two parameters control the algorithm, the number of variables in the random subset at each node and the number of trees in the forest. The choice of the number of features per node n_f and number of trees n_T may improve the classification, they are obtained by an iterative process choosing for which misclassification error rate is a minimum. Random Forest was run growing and combining 2000 trees. Results are showed in Table 5.17. These results are used throughout this section to design our Random Forest classifiers.

	PNS	PNS fresh	PNS cryo	IJPM	IJPM fresh	IJPM cryo
nb of features	1	5	1	8	1	1

Table 5.17: Minimum misclassification error rate according to the number of features, usually one or two features gives near optimum results [23]

Convergence of the error rate for the IJPM and the PNS datasets are illustrated in Figure 5.14. In both cases, to ensure convergence Random Forest is run growing and

combining 100 cycles (gray lines), and then the misclassification error rate averaged (black bold line). The generalization error rate for forest do not converge as the number of features increase, however the number of trees does [23].

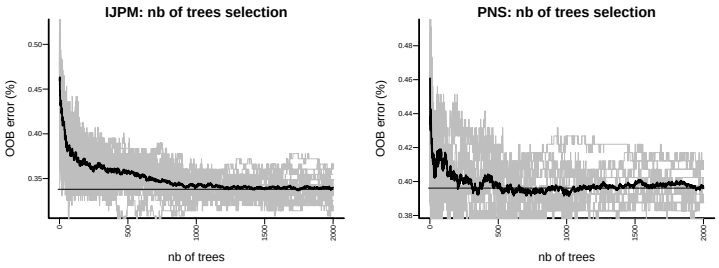


Figure 5.14: Error rate of the prediction on the IJPM and PNS data function of the number of trees; the x-th tree being the error rate for all trees up to the x-th. Error converges to 0.34 and 0.39, respectively

For the Random Forest approach, the variance of the error estimates for the PNS data is also is smaller that those based on the IJPM data certainly due to noise as seen for the latter classifiers.

In order to assess the benefits of splitting the dataset into fresh and cryo zygotes, the error obtained with the splitted data set and the error obtained with all data are compared. Within the splitted data the IJPM is better than PNS in respect to misclassification error rate (0.376 < 0.4), concerning all data together PNS is also better (0.338 < 0.394) the difference between the datasets can be considered as significant as the quartiles of IJPM and PNS in 10-fold cross-validation are well separated. Results are showed in Table 5.20. Hence, we compare e_{split}^{IJPM} with e_{all}^{IJPM}, finally $e_{all}^{IJPM} < e_{split}^{IJPM}$ (0.338 < 0.376). The smallest misclassification error with trees is obtained with the IJPM dataset. For 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $\mathbf{0.338 \pm 0.02}$.

Histograms of classification error rate are shown in Figure 5.16. Vertical gray line represents the error rate with the CPNS scoring in Table 5.3, the black line represents the mean of the error rate with the Random Forest procedure in Table 5.20. Error rate with Random Forest procedure (black) is always below the error rate with CPNS scoring (grey) even in the worst of iteration, except in the case of thawed-zygotes. Hence, the classification with Random Forest is always better even in the worst of iteration.

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
	all data		fresh		cryo	
true/perd	bad	good	bad	good	bad	good
bad	0.479 ± 0.0143	0.102 ± 0.0142	0.23 ± 0.0198	0.241 ± 0.0198	0.671 ± 0.0113	0.0231 ± 0.0113
good	0.236 ± 0.0109	0.182 ± 0.0109	0.188 ± 0.0227	0.341 ± 0.0227	0.299 ± 0.00744	0.00706 ± 0.00744

Table 5.18: Confusion matrix for 10-fold cross-validation for IJPM data

5.6.3 Conclusion

The main conclusion of this section is that the smallest misclassification error with Random Forest is obtained with the **IJPM dataset** without distinction between their

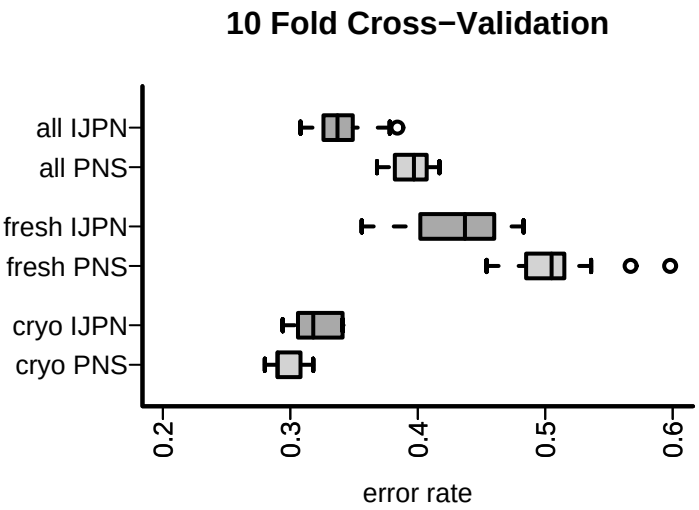


Figure 5.15: 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

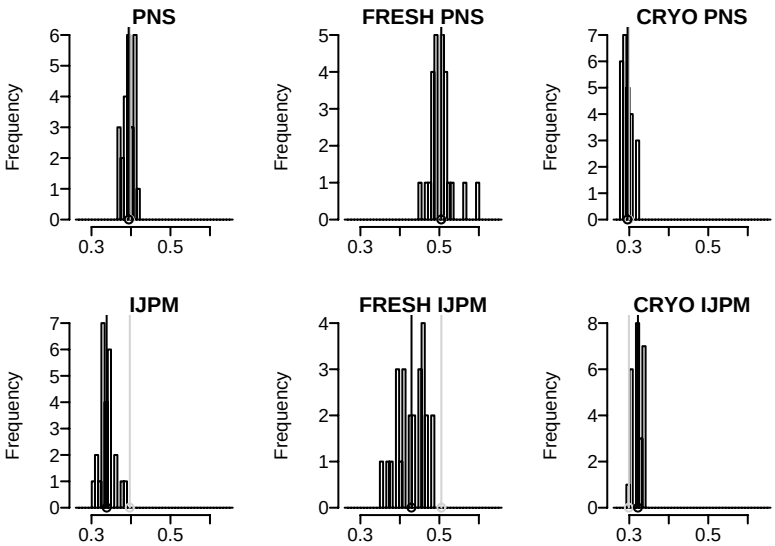


Figure 5.16: Histogram of 10-fold cross validation for both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

conservation type, its value is $\mathbf{0.338 \pm 0.02}$ for 10-fold cross-validation. Furthermore, with IJPM dataset, the classification with any grown forest is always more accurate than with CPNS.

	all data		fresh		cryo	
true \ pred	bad	good	bad	good	bad	good
	all data		fresh		cryo	
true/perd	bad	good	bad	good	bad	good
bad	0.507 ± 0.0106	0.0863 ± 0.0106	0.177 ± 0.0167	0.308 ± 0.0167	0.685 ± 0.00725	0.00673 ± 0.00725
good	0.308 ± 0.00809	0.099 ± 0.00808	0.197 ± 0.0226	0.318 ± 0.0226	0.289 ± 0.00913	0.0194 ± 0.00913

Table 5.19: Confusion matrix for 10-fold cross-validation for PNS data

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.394 ± 0.015	0.505 ± 0.03	0.296 ± 0.0127	0.4 ± 0.0158
IJPM	0.338 ± 0.0187	0.429 ± 0.0364	0.322 ± 0.0146	0.376 ± 0.0194

Table 5.20: Summary of 10-fold error rates

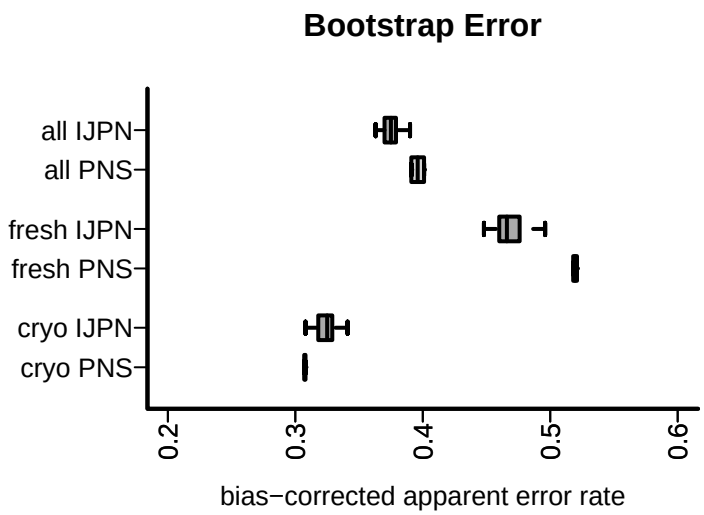


Figure 5.17: Bootstrap error rate both set of data IJPM and PNS, each set of data is represented entirely or is divided by the cryo-conservation feature

	e_{all}	e_{fresh}	e_{cryo}	e_{split}
PNS	0.396 ± 0.00707	0.52 ± 0.00212	0.308 ± 0.000707	0.414 ± 0.00141
IJPM	0.375 ± 0.00647	0.468 ± 0.0108	0.324 ± 0.00735	0.396 ± 0.00676

Table 5.21: Summary bootstrap error rates

5.6.4 Variable importance

For the purpose of understanding Random Forest "black box", internals estimates of variable importance are computed. The importance of variables changes depending on the input parameter controlling the number of variables used at each node. Results in this section are obtained by setting the parameter to the values of Table 5.17.

As IJPM dataset gives more accurate results, we focus our analysis on it. Figure 5.18 is the percent increase in misclassification rate as compared to the out-of-bag

rate. Figure 5.19 represents the variation of importance of the 10 first features. First feature represented by its boxplot is clearly detached from the others, then the decay is slow.

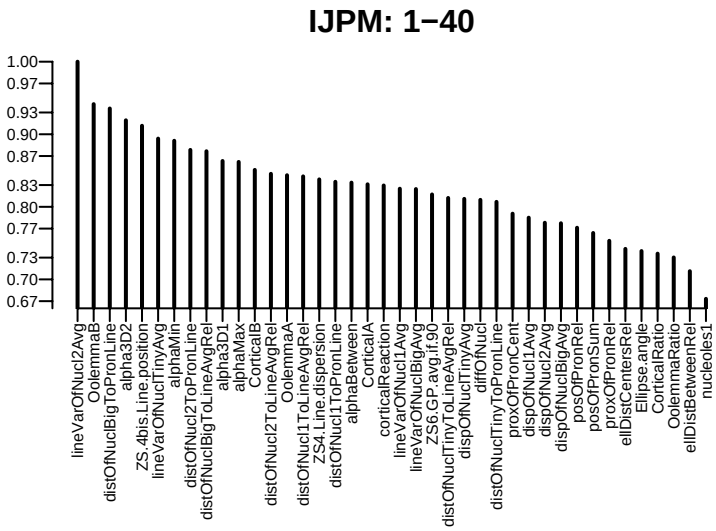


Figure 5.18: IJPM's 40 first features, the importance of each one is as percent increase in misclassification

The first feature $lineVarOfNucl2Avg$ is clearly the most important feature in respect to Random Forest classification accuracy. This measure reflects the polarization and alignment of the minority nucleoli.

We compare feature selection based on each test **Pearson**, **Kendall** and **Kolmogorov** of Section 4.2.4 and Random Forest feature selection listed in Table 5.22. Eight of ten significant variables are in the 15 first features selected with Random Forest. Two of them $diffOfNucl$ and $nucleoles2$ are in the 26-th and 50-th position, respectively.

	feature	score	pearson	kedall	kolmog.	RF position
1	diffOfNucl	1	x	x	x	26
2	OolemmaB	0.966	x	x	x	2
3	alphaMin	0.959	x	x	x	7
4	ZS.4bis.Line.position	0.939	x	x	x	5
5	nucleoles2	0.939	x	x	x	50
6	CorticalB	0.922	x	x	x	12
7	alpha3D2	0.899	–	x	x	4
8	distOfNucl2ToPronLine	0.895	x	x	x	8
9	distOfNuclBigToLineAvgRel	0.892	–	x	x	9
10	distOfNucl1ToLineAvgRel	0.875	–	x	x	15

Table 5.22: Columns are marked with a cross if the feature is significant for each test

The number of features cutoff is not clearly defined, indeed a cutoff step in Figure 5.18 is not obvious. Moreover, choosing features form Table 5.22 is not clear. Once again, this study reflects the difficulties establishing cutt-off in our features.

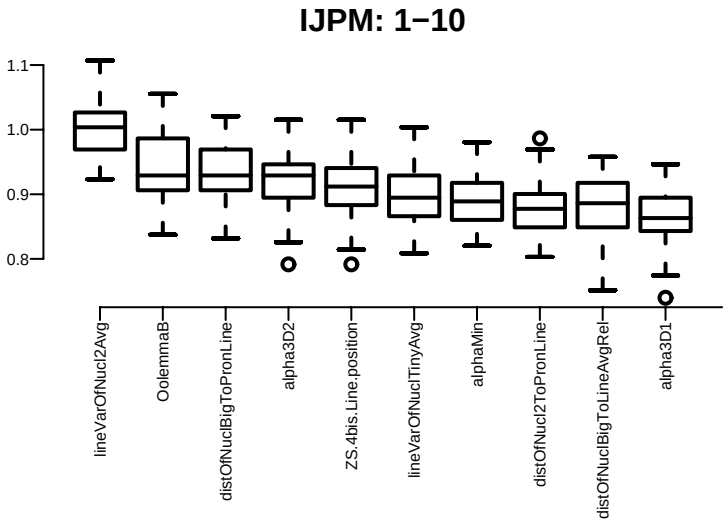


Figure 5.19: IJPM's 10 first features variation

Concerning PNS dataset, the feature *distrNucl* is totally detached from the others, we will see later that this variable is strongly correlated with the conservation type. This feature represented by the first boxplot of Figure **??**, is significantly more important than the others, indeed its quartiles are well separated.

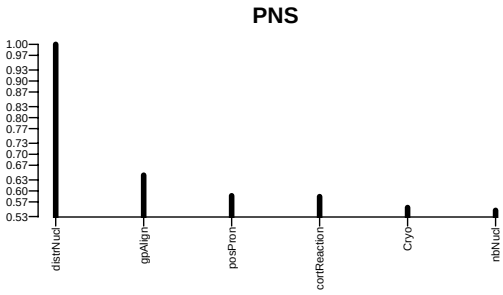


Figure 5.20: 40 first variable most important as percent increase in misclassification

5.6.5 Supervised Classification with conservation type

In Section 4.1.2, we find that the preservation state of the zygotes was strongly correlated to its success. Hence, it is interesting to find features influenced by the preservation. Thus, we use th Random Forest algorithm which is usually adapted for a measure of feature importance. We ran 100 random forest, using 2000 trees in each of them to estimate the importance represented in Figure 5.22.

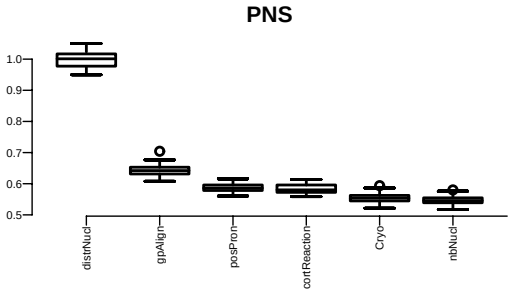


Figure 5.21: Variable importance as percent increase in misclassification

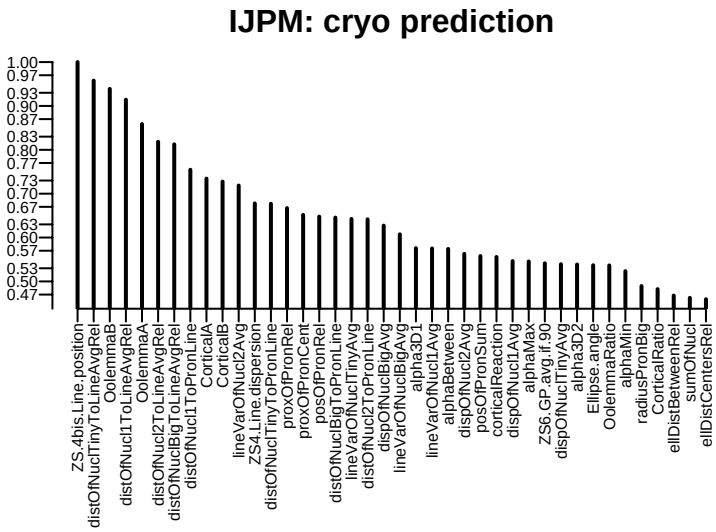


Figure 5.22: Variable importance as percent increase in misclassification for cryo-preserved zygotes

First seven of ten features are in relation to the polarisation of the nucleoli (see Section 2.1). We can conclude, that this measure is influent to the classification of the preservation type, at least with our dataset. Hence, if we examine these distributions a little bit closer, our measures tells us that **within nucleoli dispersions and dispersion in respect to the "pronuclei line" (see Secction 2.1) increases when the zygotes were cryo-preserved** (see Figure 5.23). Furthermore, in these ten first features two of them are measurements of Oolemma "radius" (Oolemma is fitted to an ellipse), Oolemma is smaller for cryo-preserved zygotes.

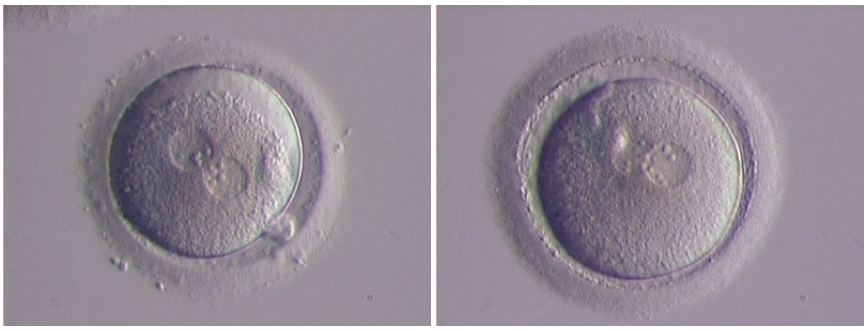


Figure 5.23: Fresh (left image) and Frozen-thawed zygote (right image), nucleoli dispersions increases when the zygotes were cryo-preserved

Chapter 6

Improvement

The goal of this project is to improve the implantation rate of the zygotes that is currently about 16% for fresh and 10% for frozen zygotes when 8-10 zygotes are collected and fertilized. The misclassification rates computed in our analysis of the classifiers is not representative for the real improvement that can be achieved by using a better classifier. An improvement of 6% in the misclassification rate (CPNS compared to IJPM) affects to the selection of a single zygote. However, the implantation rate is the output of a much more complex process. First, two zygotes must be chosen from a set of 8 to 10 fresh zygotes (the rest of the zygotes are frozen). If these two zygotes failed to lead to pregnancy, then other two zygotes must be chosen among the frozen ones. This process is repeated until success or there are no more zygotes. An improvement of 6% in the misclassification rate affects to the selection of each zygote. Therefore, the expected gain in the probability of success should be larger than 6%. Furthermore, to accurately compute this probability, the probability of success of a fresh zygote and a frozen zygote must be taken into account. In this section we develop the theory that allows computing the probability of success of the whole process. As an exercise, we start our discussion with a strategy in which only one zygote is selected at a time. This will serve us as basis to develop the more complicated theory for the case of selecting (as is done in practice) two zygotes at a time.

Selection of one zygote at a time

In this section we study the case in which we first select a single fresh zygote and freeze the rest. If it does not lead to pregnancy, then we take another one from among the frozen ones and repeat this procedure until one of the zygotes succeed or there no more zygotes. We assume that there is initially N fresh zygotes. We refer to fresh zygotes with the label F and to cryo zygotes with label C. We refer to zygotes actually leading to pregnancy with the label G (Good zygotes) and to zygotes actually not leading to pregnancy with the label B (Bad zygotes). Independently of whether a zygote is good or bad, the classifier classifies it as good or bad. We refer to zygotes classified as good with the label g and to zygotes classified as bad with the label b. For the sake of clarity, at some points we start our discussion with three zygotes and, subsequently, we generalize to N zygotes.

Each fresh zygote has a probability $p(G|F)$ and $p(B|F) = 1 - p(G|F)$ of being Good or Bad (correspondingly, $p(G|C)$ and $p(B|C)$ when they are frozen). These

probabilities are assumed to be known by previous experience. Our analysis of the confusion matrix of each classifier give us the probabilities of being classified with a label when the zygote actually have some other label. These probabilities are different depending on whether the zygotes are fresh or not. For instance, the confusion matrix gives us information about the probability that a Good, fresh zygote is classified as good, that is, $p(g|G \cap F)$. Table 6.1 summarizes all the probabilities provided by the confusion matrices. Note that given a zygote that is actually Good, it must hold that $p(g|G \cap F) + p(b|G \cap F) = 1$. That means that all rows of the confusion matrices must add up to 1.

	classified as bad	classified as good
actually Bad	$p(b\|B \cap F)$	$p(g\|B \cap F)$
actually Good	$p(b\|G \cap F)$	$p(g\|G \cap F)$

	classified as bad	classified as good
actually Bad	$p(b\|B \cap C)$	$p(g\|B \cap C)$
actually Good	$p(b\|G \cap C)$	$p(g\|G \cap C)$

Table 6.1: Probabilities provided by the confusion tables.

The first zygote in our case is selected among N fresh zygotes. Thus, the probability of success at the first zygote can be computed as follows.

Given any fresh zygote, there are four possibilities: either it is Good and it is classified as good (this will occur with probability $p(G \cap g|F)$), or it is Good and it is classified as bad ($p(G \cap b|F)$), or it is Bad and it is classified as good ($p(B \cap g|F)$), or it is Bad and it is classified as bad ($p(B \cap b|F)$). These probabilities can be easily computed with the data available (confusion matrices and the probability of a fresh zygote of being Good). For instance, $p(G \cap g|F) = p(g|G \cap F)p(G|F)$. If a zygote is classified as bad (either if it is actually Good or Bad), it will not be selected. Thus, we are left only with three possibilities: it is Good and it is classified as good ($p(G \cap g|F)$), or it is Good and it is classified as bad ($p(G \cap b|F)$), or it is classified as bad($p(b|F) = p(b|G \cap F)p(G|F) + p(b|B \cap F)p(B|F)$).

Given three zygotes, we will achieve success at the first choice if at least one zygote is Good and it is classified as good. Thus, there are only six possible successful cases summerized in Table 6.2 where $\#Gg|F$ represents the number of fresh zygotes that are actually Good and are classified as good. For instance, the first case indicates that the three of them are Good and classified as good (success is guaranteed). Int the second case two of them are Good and classified as good and the other one is Bad but classified as good. In this case, success is achieved with probability 2/3 since we have three zygotes classified as good but only two are really Good. In the last case, two of the zygotes are classified as bad, so they will not be chosen (independently of whether they are actually Good or Bad). The only zygote that is classified as good is actually Good and, for this reason, success is guaranteed.

In general, the probability of success given a certain combination of zygotes $(\#Gg|F, \#Bg|F, \#b|F)$ can be easily computed as

$$p(success|\#Gg|F, \#Bg|F, \#b|F) = \frac{\#Gg|F}{\#Gg|F + \#Bg|F} \tag{6.1}$$

We will achieve success at the first zygote if we are in any of the situations in Table 6.2. Thus, we need to compute the probability of encountering each case. These

$\#Gg\|F$	$\#Bg\|F$	$\#b\|F$	$p(success\|\#Gg\|F,\#Bg\|F,\#b\|F)$
3	0	0	1
2	1	0	$\frac{2}{3}$
2	0	1	1
1	2	0	$\frac{1}{3}$
1	1	1	$\frac{1}{2}$
1	0	2	1

Table 6.2: Probability of success at the first choice.

probabilities in turn depend on the probability of a single zygote of being actually Good and classified as good, of being actually Bad and classified as good, or of being classified as bad ($p(G\cap g|F)$,$p(B\cap g|F)$,$p(b|F)$).

The first case (($\#Gg|F=3,\#Bg|F=0,\#b|F=0$)) occurs with probability $p(G\cap g|F)^3 p(B\cap g|F)^0 p(b|F)^0$ since we need the three zygotes be of the same type. We represent this case by (Gg,Gg,Gg) meaning that the first zygote is actually Good and classified as good (Gg), the second is also Gg and so is the third. Thus, we can see there is a single combination of zygote labels leading ((Gg,Gg,Gg)) to the first case. However, we can find three label combinations leading to the second case: (Gg,Gg,Bg), (Gg,Bg,Gg), (Bg,Gg,Gg). Thus, its probability is $3p(G\cap g|F)^2 p(B\cap g|F)^1 p(b|F)^0$. In general, the probability of each case follows a multinomial distribution whose general term is

$$\begin{gathered} p(\#Gg|F,\#Bg|F,\#b|F|N) = \\ \binom{N}{\#Gg|F}\binom{N-\#Gg|F}{\#Bg|F}\binom{N-\#Gg|F-\#Bg|F}{\#b|F} p(G\cap g|F)^{\#Gg|F} p(B\cap g|F)^{\#Bg|F} p(b|F)^{\#b|F} \end{gathered} \tag{6.2}$$

Finally, with $N=3$ zygotes, the probability of succeeding at the first zygote can be computed as

$$\begin{aligned} p(success|N=3,F) \quad = \quad & p(success|3,0,0)p(3,0,0|3)+p(success|2,1,0)p(2,1,0|3)+ \\ & p(success|2,0,1)p(2,0,1|3)+p(success|1,2,0)p(1,2,0|3)+ \\ & p(success|1,1,1)p(1,1,1|3)+p(success|1,0,2)p(1,0,2|3), \end{aligned} \tag{6.3}$$

and, in general,

$$\begin{gathered} p(success|N,F) = \\ \sum_{\#Gg|F=1}^{N}\sum_{\#Bg|F=1}^{N-\#Gg|F} p(success|\#Gg|F,\#Bg|F,\#b|F)p(\#Gg|F,\#Bg|F,\#b|F|N), \end{gathered} \tag{6.4}$$

where $\#b|F = N-\#Gg|F-\#Bg|F$.

Finally, if we have $N=3$ zygotes, our current strategy chooses a fresh one. If it fails to lead to pregnancy, we choose a cryo zygote. If it fails, we take the third zygote. The probability of succeeding at choice 1 is, we have already proved it, $p(success|N=3,zygote=1)=p(success|3,F)$. Therefore, we fail with probability $p(failure|N=3,zygote=1)=1-p(success|N=3,zygote=1)$. Thus, the probability of succeeding at zygote number 2 (that was frozen) is $p(success|N=3,zygote=2)=p(failure|N=3,zygote=1)p(success|N=2,C)$. Note that the probability of succeeding at zygote number 2 is the probability of failing at zygote number 1 and the succeeding with a cryo zygote among 2 (because one of the 3, the fresh one, has already

failed). Finally, if zygote number 2 fails, the probability of succeeding at zygote number 3 is $p(success|N=3, zygote=3) = p(failure|N=3, zygote=1)p(failure|N=2, zygote=2)p(success|N=1, C)$. Table 6.3 summarizes this reasoning.

Egg number i	$p(success\|N=3, zygote=i)$
1	p(success|3,F)
2	(1-p(success|3,F))p(success|2,C)
3	(1-p(success|3,F))(1-p(success|2,C))p(success|1,C)

Table 6.3: Probability of success selecting one zygote at a time.

In general, the probability of succeeding at zygote number i is

$$p(success|N,i) = \begin{cases} p(success|N,F) & i=1 \\ (1-p(success|N,F))\left(\Pi_{j=2}^{i-1}(1-p(success|N-j+1,C))\right)p(success|N-i+1,C)0 & i \neq 1 \end{cases} \tag{6.5}$$

Finally, with N zygotes, the overall probability of success of this strategy is the sum of the probabilities of succeeding at any zygote between 1 and N, that is,

$$p(success|N) = \sum_{i=1}^{N} p(success|N,i) \tag{6.6}$$

Selection of two zygotes at a time

Our previous analysis addresses a strategy that is not the currently followed in practice. Instead of selecting a single fresh zygote and freezing the rest, the current strategy works with pairs of zygotes. First, a pair of fresh zygotes is chosen and the rest are frozen. If both zygotes fail, then other two zygotes are thawed and implanted. If they fail too, other two zygotes are defrozen. This procedure is iterated until success or there are no more zygotes.

Although selecting a single zygote at a time is different from selecting a pair, our previous analysis helps calculating the probability of success of the actual implantation strategy. Equations 6.5 and 6.6 remain the same in the case of selecting two zygotes if instead of interpreting i as the number of the zygote, i is the number of the pair and N is the total number of pairs. The main difference between selecting pairs and selecting single zygotes comes at Equation 6.4. Given N pairs of fresh zygotes, the probability of succeeding taking a pair of zygotes is given by

$$p_2(success|N,F) =$$

$$p_1(success|2N,F) + p_1(success|2N-1,F) - p_1(success|2N,F)p_1(success|2N-1,F) \tag{6.7}$$

where $p_2(success|N,F)$ is the probability of succeeding after selecting a pair among $2N$ zygotes, and $p_1(success|M,F)$ is the probability of succeeding after selecting a single zygote among M zygotes (this probability is the one given in Equation 6.4).

Results

The probability of obtaining a Good zygote if it is fresh is $p(G|F) = 0.17$ while if it is thawed it drops to $p(G|C) = 0.1$[1]. With these probabilities at hand and the confusion

[1]Data from the *LABR* center, 2003

matrices corresponding to each of the classifiers studied in this report, we compute the probability of having a pregnancy with our best classifier Random Forest for IJPM, and SVM for PNS, finally we compare it to the CPNS. Results are summarized in Figure 6.1. The probability of success after 4 attempts (i.e. 8 zygotes) for CPNS (human expert visual inspection of six zygotes) is of **0.308**, for PNS dataset it is of **0.582** and for IJPM dataset it is of **0.656**. We also compute PNS with Random Forest and obtain **0.388**.

With IJPM dataset and Random Forest, the first step represents an improvement in respect to CPNS of 21.4%. Improvement in the following step is of 7.4%. As state above, the gain in the probability of success is larger than the improvement of the classification error rate. PNS dataset with SVM also leads to an improvement in respect to CPNS of 9.6%, the following step imprvement is of 9.4%.

```
CPNS - 15 Threshold -
P(success at pair 1)=0.262496
P(success at pair 2)=0.0239758
P(success at pair 3)=0.0148499
P(success at pair 4)=0.00626926
Total probb of success = 0.307591
```

```
PNS - Random Forest -
P(success at pair 1)=0.304239
P(success at pair 2)=0.0450919
P(success at pair 3)=0.0274029
P(success at pair 4)=0.0114865
Total probb of success=0.38822
```

```
PNS - SVM -
P(success at pair 1)=0.359229
P(success at pair 2)=0.118515
P(success at pair 3)=0.0722277
P(success at pair 4)=0.0316486
Total probb of success=0.58162
```

```
IJPM - Random Forest -
P(success at pair 1)=0.47636
P(success at pair 2)=0.0975371
P(success at pair 3)=0.0575458
P(success at pair 4)=0.0244482
Total probb of success=0.655891
```

Table 6.4: Probability of success computations applying CPNS, SVM to PNS dataset and Random Forest to PNS and IJPM datasets

Four groups of vertical lines are represented in Figure 6.1, each group represents a transfer procedure. Although, this procedure is iterated until success or there are no more zygotes, we fix four iterations. Within each data set (IJPM and PNS), implantation rates increase rapidly, then levels off. IJPM is always above CPNS and PNS, indicating a significant improvement. Classification with IJPM dataset is clearly more accurate.

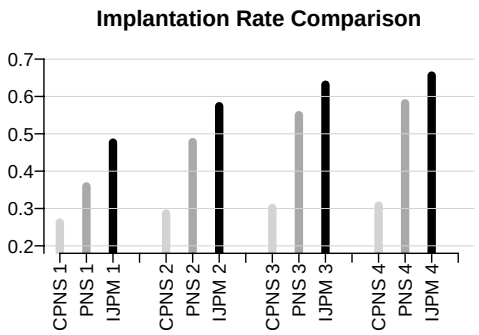


Figure 6.1: Implantation rate comparison after four transferts

Chapter 7

Conclusion

Our aim in this study is to make morphometric measurements and observations based on digital images of zygotes in order to predict the likely outcome. To achieve this two methods are available: one of them is to use a plug-in for *ImageJ* developed at the *EPFL* in collaboration with the *LABR* Laboratories. The collected dataset is labeled as **IJPM** for **I**mageJ **P**ronuclear Morphometric **M**easurements. The other method is a visual inspection of six zygotes characteristics labeled **PNS** for **P**ronuclear **S**core, each characteristic can take values from 1 to 3 (from worst to best).

In this report, we study the power of each individual feature to classify the "zygote class". For doing this, we measure the degree of association between each feature and the "zygote class". The degree of association is measured differently depending on the nature of the feature: categorical or continuous. Categorical data were, type of conservation of the zygotes fresh and frozen and insemination type IVF [1] or ICSI [2], we use the Chi-Square test which evaluates the independence of two variables, since its assumptions were fulfilled. Our analysis shows that the insemination type does not affect the class of the zygote ($\chi^2 = 0.015$ and $p - value = 0.9011$ not significant). On the other hand, the conservation state of the zygotes is strongly correlated to its success ($\chi^2 = 7.333$ for $p - value < 0.01$ significant). In fact, the difference between those two classes were so large, that in our classification studies we also had to analyze zygotes separately depending on whether they were fresh or not.

Concerning continuous features, in principle the degree of association is measured with Pearson product-moment correlation. However, the assumptions of this measure were violated by our data. Alternatively, we used two other measures of association, Kendall's Tau correlation and Kokmogorov-Smirnov goodness-of-fit. Seven features were significant in respect to the class "good" or "bad", for each test Pearson, Kendall and Kolmogorov.

In order to measure the contribution of each feature to the overall variance, we made use of PCA. PCA was computed using the correlation matrix, as we wanted to give to all features the same weight. The decrease in contributions from the principal components were relatively low. To obtain 99% of the overall variance, 37 first components had to be used; with ten components we had only 52% of the overall variance (see Figure **??**). We cannot conclude that any feature has a strong contribution to the overall variance. Furthermore, significant variables obtained with the degree of association tests, were widespread on the PCA axis. This means that they are not too much

[1] In vitro fertilisation
[2] Intra-Cytoplasmic Sperm Injection

correlated among them and, therefore, they bring enough information about the data set.

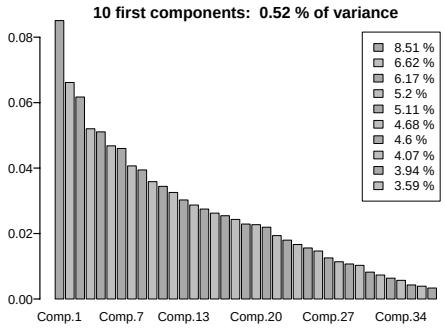


Figure 7.1: Relative variance of the 37 first components, 99% of the overall variance

Then, we compared results obtained with several well-known classification methods, to identify zygotes with high implantation potential based on PNS and IJPM datasets. To compare the performance of the classifiers cross-validation within the original data set and bootstrap techniques were used to provide a nearly unbiased estimate.

In order to assess the benefits of splitting the dataset into fresh and cryo zygotes when classifying, the error obtained with the divided data set and the error obtained with all data were compared. The error for a given classifier using all data was referred to as e_{all}, while the error with the splitted dataset was called e_{split}.

LDA is a classical statistical approach for predicting samples of unknown classes, based on training samples with known classes. LDA tries to provide more class separability and draw a decision region between the given classes maximizing the ratio of between-class variance to the within-class variance in any particular dataset guaranteeing maximal separability. CPNS scoring was seen as an LDA classification with an unitary weight vector, the threshold was already established by the laboratory to 15. This selection was confirmed by computing the implantation rate as function of the CPNS (see Figure 7.2). Indeed, the implantation rate increases for all zygotes when the cumulated pronuclear score is greater than 8. At level 15 the implantation rate is maximum.

The smallest misclassification error with equiweighted LDA (CPNS) is obtained without distinction between their conservation type, its value is $e_{all} = \mathbf{0.398}$ for 10-fold cross-validation. For frozen-thawed zygotes the accuracy of the classifier was very low. In fact, all frozen zygotes were classified as bad. However, the overall classifier accuracy was of 0.3, this fact was explained by the small number of thawed-zygotes in our sample, only 26 of 59 ($26/(26+59) \approx 0.3$) were good. Even if we compare all our classifiers with this accuracy, this fact was taken in account.

The whole IJPM dataset performs better classification than PNS for conventional LDA in respect to the estimation of the error rate. The mean error estimation for IJPM without distinction between their conservation type was $e_{all} = \mathbf{0.36} \pm \mathbf{0.02}$ whereas the mean for PNS was 0.39 ± 0.01. However, the variance of the error estimates for the PNS data was smaller that those based on the IJPM data due to noise (PNS took only 3 values, whereas values of IJPM were continuous).

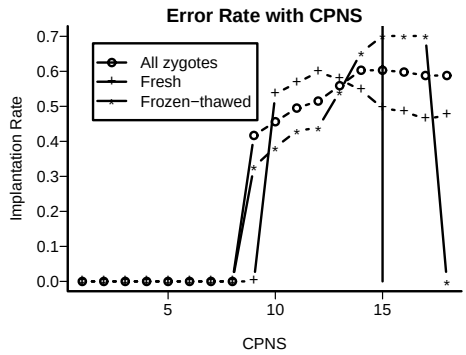


Figure 7.2: Implantation rate as function of the cumulated pronuclear score (CPNS)

Trees classifiers lead to an overall higher misclassification rate compared to LDA, furthermore trees showed lower stability, interquartile ranges were broader than those previously computed with LDA. Therefore, our preliminary analysis suggested that trees in our case were less useful than LDA. However, trees are a useful method to summarize data and to make a quick human diagnostic. Indeed, two quick human diagnostics can be made before the transfer, the first is to calculate the CPNS score and the second is to drop the zygote down the tree of Figure 7.3. The advantage of using our tree classifier is that a zygote with a score under 15 can be considered as any zygote. Hence, allowing a classification with all zygotes as opposed to the CPNS classification. This advantage is that in our sample, only 10% zygotes have a score over or equal to 15. The smallest misclassification error with tree was obtained with the PNS dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value was of $e_{all} = \mathbf{0.41} \pm \mathbf{0.02}$ for 10-fold cross-validation.

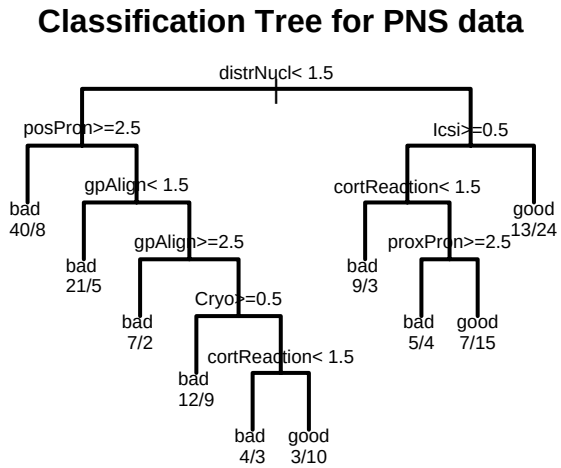


Figure 7.3: Classification Tree for PNS data

Support vector machine (SVM) model is a nonlinear discriminant analysis. The basic idea of SVM is to seek for a separating non-linear decision boundary maximizing a margin between two classes with a higher dimensional feature space. SVM approach leaded to an overall lower misclassification rate compared to LDA, however, it's stability was also lower. The smallest misclassification error with SVM was obtained with the IJPM dataset, for 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value was of $e_{all} = \mathbf{0.378} \pm \mathbf{0.02}$ for 10-fold cross-validation.

Random Forest is a classifier that is built from multiple trees generated from random sampling of the observations, and the features. One of the reason for performing random forest to our data was that random forest is robust against over-fitting, and it is more robust with respect to noise than other classification methods [23]. And as seen with PCA, noisy IJPM features had a strong contribution to the overall variance for the IJPM dataset. For the purpose of understanding Random Forest "black box", internals estimates of variable importance were computed. Handling interactions among variables is another advantage compared to LDA.

The smallest misclassification error with Random Forest is obtained with the IJPM dataset. For 10-fold cross-validation and bootstrap techniques, without distinction between their conservation type, its value is of $e_{all} = \mathbf{0.338} \pm \mathbf{0.02}$. Difference between the datasets can be considered as significant as the quartiles of IJPM and PNS in 10-fold cross-validation are well separated.

Overall, we have found that the Random Forest approach both leads to an overall lower misclassification rate as well as to a more stable assessment of classification errors compared with the developed methods (additional methods can be neural networks, k-nearest neighbor classifier, ...). Therefore, our preliminary analysis suggest that Random Forest may be more useful than other methods to classify samples based on digital images. However, the misclassification rates computed in our analysis of the classifiers is not representative of the real improvement that can be achieved by using a better classifier. Hence, we develop the theory that allows to compute the probability of success of an implantation process. Another reason of developing this theory is that our sample is not a random sample, in fact we chose the zygotes leading to multiple pregnancies, and therefore the sampling distributions are biased, this bias is also corrected by this procedure.

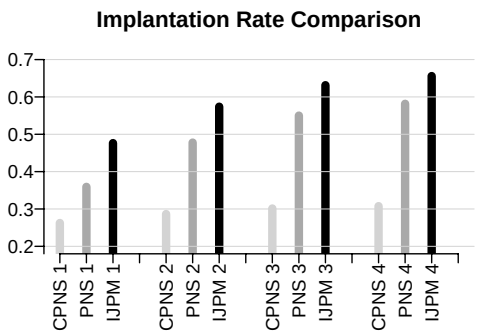


Figure 7.4: Implantation rate comparison after four transferts

We compute the probability of having a pregnancy with our best classifier Random Forest for IJPM, and SVM for PNS, finally we compare it to the CPNS. The probability of success after 4 attempts (i.e. 8 zygotes) for CPNS is of **0.308**, for PNS dataset it is of **0.582** and for IJPM dataset it is of **0.656**.

With IJPM dataset and Random Forest, the first step represents an improvement in respect to CPNS of 21.4%. Improvement in the following step is of 7.4%. As state above, the gain in the probability of success is larger than the improvement of the classification error rate. PNS dataset with SVM also leads to an improvement in respect to CPNS of 9.6%, the following step improves to 9.4%.

Four groups of vertical lines are represented in Figure 7.4, each group represents a transfer procedure. Although, this procedure is iterated until success or there are no more zygotes, we fix four iterations. Within each data set (IJPM and PNS), implantation rates increase rapidly, then levels off. IJPM is always above CPNS and PNS, indicating a significant improvement. Classification with IJPM dataset is clearly more accurate.

No study has attempted to evaluate the contribution of morphological characteristics of zygotes automatically detected by an advanced image analysis tool, these characteristics can be used as markers of future embryo developmental competence using statistical tools. Statistical tools allow us to transfer only the embryos classified as "good" while minimizing the high order multiple pregnancies. However, to assess our results theses experiments must be done with more data.

Aknowledgements

I wish to acknowledge my supervisor: Dr. Carlos Oscar Sánchez Sorzano at the "Consejo Superior de Investigaciones Científicas" (CSIC) who helped me refine my project goals and for his constructive comments. I also thank Dr. Alfred Senn and Dr. Françoise Urner at "Laboratoire d'Andrologie et de Biologie de la Reproduction" (LABR) for their active support in developing this project. I am, of course, indebted to the Swiss Federal Institute of Technology (EPFL) , for providing this project.

Bibliography

[1] F. Puissant, M. Van Rysselberge, P. Barlow, J. Deweze, and F. Leroy, "Embryo scoring as a prognostic tool in IVF treatment," *Human Reproduction*, vol. 2, pp. 705–8, 1987.

[2] C. Giorgetti, P. Terriou, P. Auquier, E. Hans, J. L. Spach, J. Salzmann, and et al, "Embryo score to predict implantation after in-vitro fertilization: based on 957 single embryo transfers," *Human Reproduction*, vol. 10, pp. 2427–31, 1995.

[3] S. Ziebe, K. Petersen, S. Lindenberg, A. G. Andersen, and A. N. Andersen A. Gabrielsen, "Embryo morphology or cleavage stage: how to select the best embryos for transfer after in-vitro fertilization," *Human Reproduction*, vol. 2, pp. 1545–9, 1997.

[4] D. Sakkas, F. Urner, D. Bizzaro, G. Manicardi, P. G. Bianchi, Y. Shoukir, and et al, "Sperm nuclear DNA damage and altered chromatin structure: effect on fertilization and embryo development," *Human Reproduction*, vol. 13 Suppl 4, pp. 11–9, 1998.

[5] A. Salumets, C. Hyden-Granskog, S. Makinen, A. M. Suikkari, A. Tiitinen, and T. Tuuri, "Early cleavage predicts the viability of human embryos in elective single embryo transfer procedures," *Human Reproduction*, vol. 18, pp. 821–5, 2003.

[6] D. K. Gardner, M. Lane, J. Stevens, T. Schlenker, and W. B. Schoolcraft, "Blastocyst score affects implantation and pregnancy outcome: towards a single blastocyst transfer," *Fertil Steril*, vol. 73, pp. 1155–8, 2000.

[7] L. A.Scott and S. Smith, "The successful use of pronuclear embryo transfers the day following oocyte retrieval," *Human Reproduction*, vol. 13, pp. 1003–13, 1998.

[8] J. D. Fisch, H. Rodriguez, R. Ross, G. Overby, and G. Sher, "The Graduated Embryo Score (GES) predicts blastocyst formation and pregnancy rate from cleavage-stage embryos," *Human Reproduction*, vol. 16, pp. 1970–5, 2001.

[9] J. Tesarik and E. Greco, "The probability of abnormal preimplantation development can be predicted by a single static observation on pronuclear stage morphology," *Human Reproduction*, vol. 14, pp. 1318–23, 1999.

[10] M. Ludwig, B. Schopper, S. Al-Hasani, and K. Diedrich, "Clinical use of a pronuclear stage score following intracytoplasmic sperm injection: impact on pregnancy rates under the conditions of the German embryo protection law," *Human Reproduction*, vol. 15, pp. 325–9, 2000.

[11] A. Salumets, C. Hyden-Granskog, A. M. Suikkari, A. Tiitinen, and T. Tuuri, "The predictive value of pronuclear morphology of zygotes in the assessment of human embryo quality," *Human Reproction*, vol. 16, pp. 2177–81, 2001.

[12] D. Sage and M. Unser, "Teaching Image-Processing Programming in Java," *IEEE Signal Processing Magazine*, vol. 20, no. 6, pp. 43–52, 2003, Using "Student-Friendly" ImageJ as a Pedagogical Tool.

[13] A. Senn, C. Vozzi, A. Chanson, P. De Grandi, and M. Germond, "Prospective randomized study of two cryopreservation policies avoiding embryo selection: the pronucleate stage leads to a higher cumulative delivery rate than the early cleavage stage," *Fertil Steril*, vol. 74, pp. 946–52, 2000.

[14] Andrew R. Webb, *Statistical Pattern Recognition*, John Wiley and Sons Ltd., 2 edition, 2002.

[15] P. Royston, "Algorithm AS 181: The W Test for Normality," *Applied Statistics*, vol. 31, pp. 176–180, 1982.

[16] P. Royston, "A Remark on Algorithm AS 181: The W Test for Normality," *Applied Statistics*, vol. 44, pp. 547–551, 1995.

[17] P. Sprent, *Data Driven statistical methods*, Chapman & Hall, London, 2 edition, 1998.

[18] F. Yates, "Contingency tables involving small numbers and the chi-square test," *Journal of the Royal Statistical Society*, vol. 1, pp. 217–235, 1934.

[19] David J. Sheskin, *Handbook of Parametric and Nonparametric Statistical Procedures*, CRC Press, LLC, 3 edition, 2004.

[20] C. Goutte, "Note on free lunches and cross-validation," *Neural Computation*, vol. 9, pp. 1211–1215, 1997.

[21] L. Breiman and P. Spector, "Submodel selection and evaluation in regression: the X random case," *International Statistical Review*, vol. 60, pp. 291–319, 1992.

[22] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*, Wadsworth & Brooks, 1984.

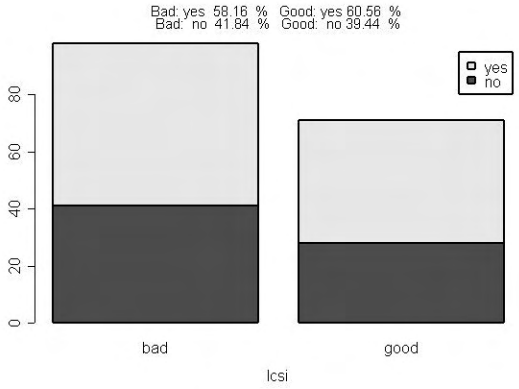
[23] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, 2001.

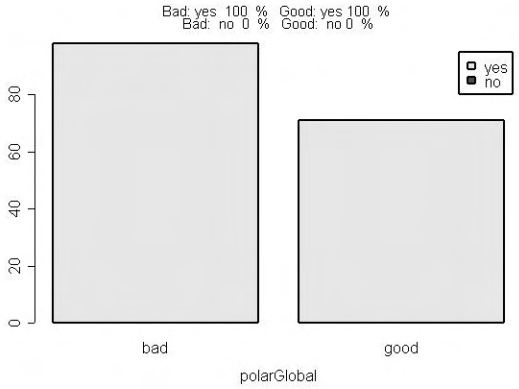
marks the start of additional material in your book. After this command chapters will be numbered with letters.

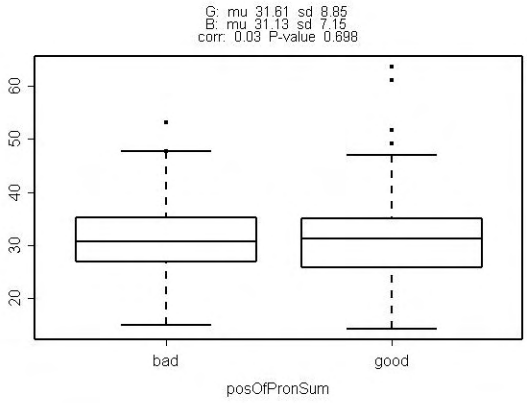
Appendix A

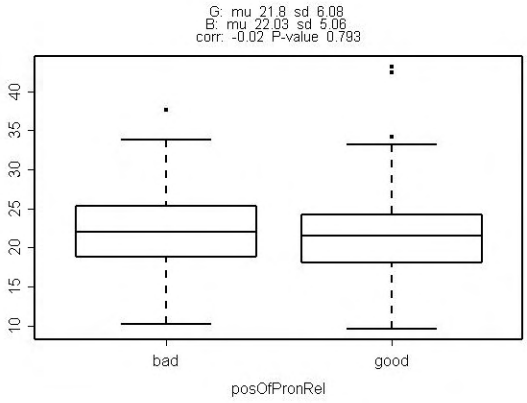
Boxplots class vs variables

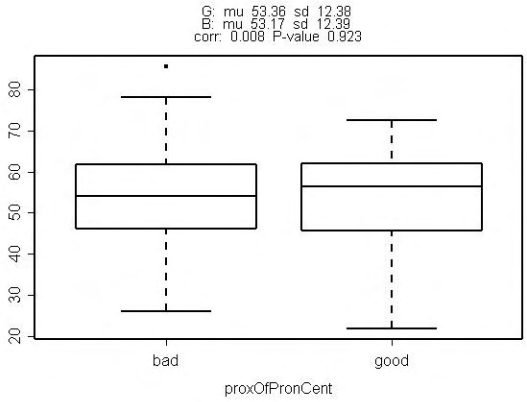
advantages:	compact, concise
disadvantages:	emphasize tails of distribution,
	discard many finer features distribution

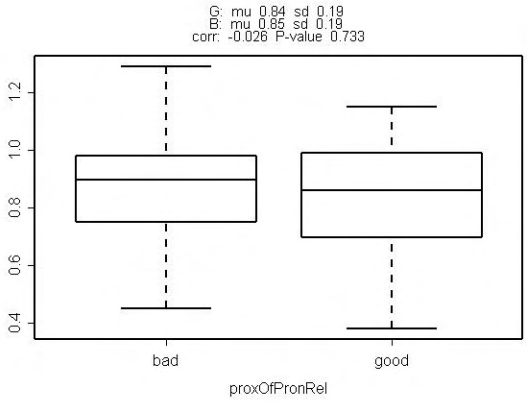

Bad: yes 58.16 % Good: yes 60.56 %
Bad: no 41.84 % Good: no 39.44 %
yes
no
0
20
40
60
80
bad
good
lcsi

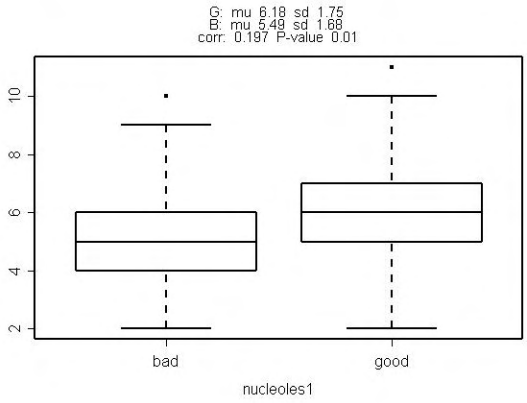

Bad: yes 100 % Good: yes 100 %
Bad: no 0 % Good: no 0 %
yes
no
0
20
40
60
80
bad
good
polarGlobal

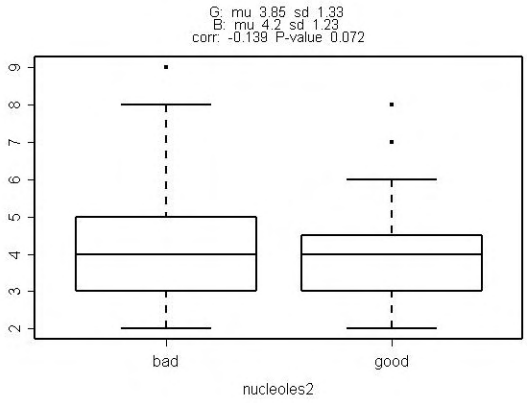

G: mu 31.61 sd 8.85
B: mu 31.13 sd 7.15
corr: 0.03 P-value 0.698
20
30
40
50
60
bad
good
posOfPronSum

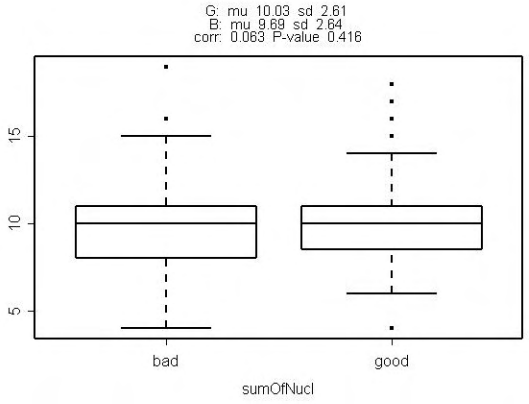

G: mu 21.8 sd 6.08
B: mu 22.03 sd 5.06
corr: -0.02 P-value 0.793
10
15
20
25
30
35
40
bad
good
posOfPronRel

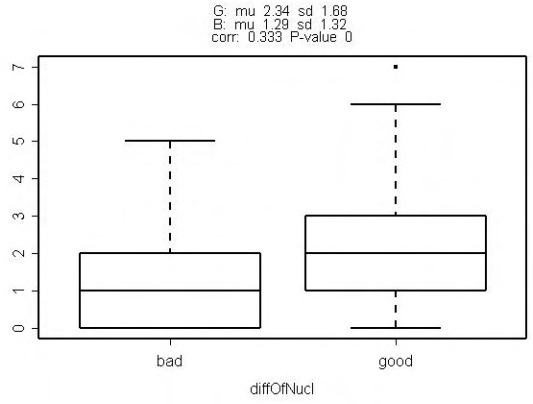

G: mu 53.36 sd 12.38
B: mu 53.17 sd 12.39
corr: 0.008 P-value 0.923
20
30
40
50
60
70
80
bad
good
proxOfPronCent

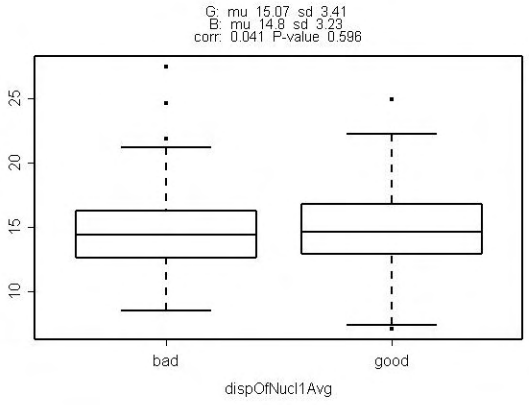

G: mu 0.84 sd 0.19
B: mu 0.85 sd 0.19
corr: -0.026 P-value 0.733
0.4
0.6
0.8
1.0
1.2
bad
good
proxOfPronRel

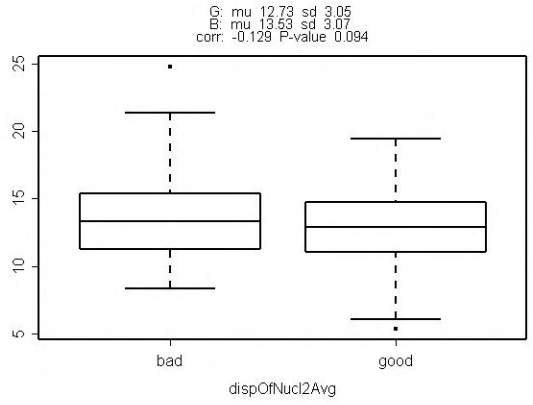

G: mu 6.18 sd 1.75
B: mu 5.49 sd 1.68
corr: 0.197 P-value 0.01
2
4
6
8
10
bad
good
nucleoles1

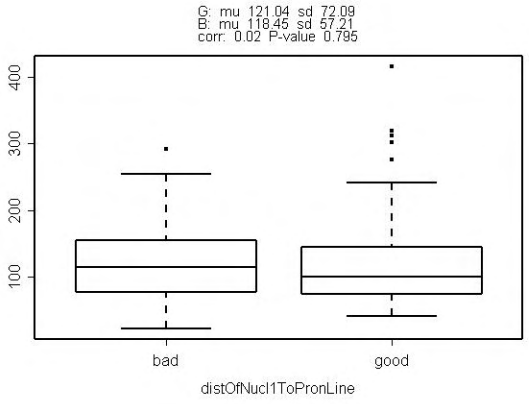

G: mu 3.85 sd 1.33
B: mu 4.2 sd 1.23
corr: -0.139 P-value 0.072
2
3
4
5
6
7
8
9
bad
good
nucleoles2

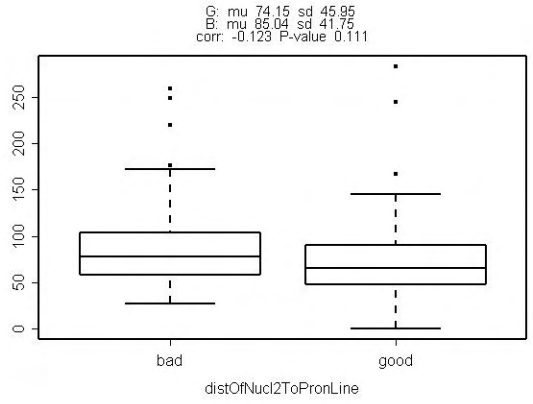

G: mu 10.03 sd 2.61
B: mu 9.69 sd 2.64
corr: 0.063 P-value 0.416
5
10
15
bad
good
sumOfNucl

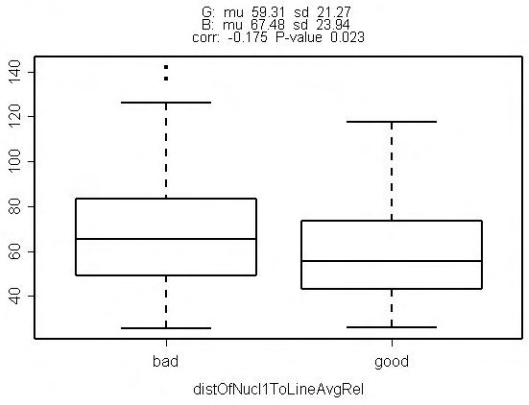

G: mu 2.34 sd 1.68
B: mu 1.29 sd 1.32
corr: 0.333 P-value 0
0
1
2
3
4
5
6
7
bad
good
diffOfNucl

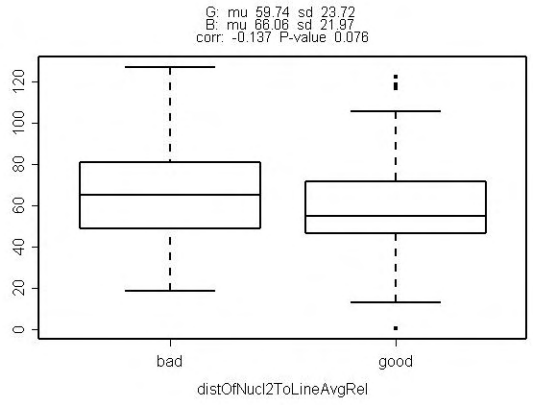

G: mu 15.07 sd 3.41
B: mu 14.8 sd 3.23
corr: 0.041 P-value 0.596
10
15
20
25
bad
good
dispOfNucl1Avg

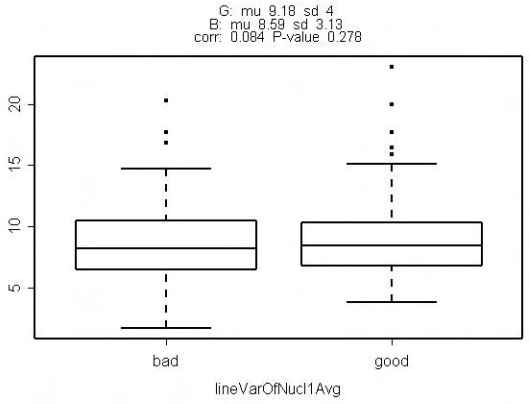

G: mu 12.73 sd 3.05
B: mu 13.53 sd 3.07
corr: -0.129 P-value 0.094
5
10
15
20
25
bad
good
dispOfNucl2Avg

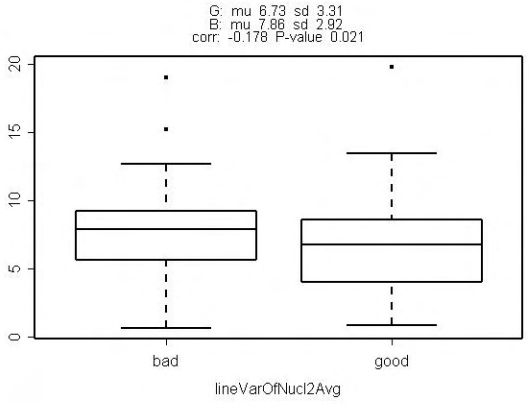

G: mu 121.04 sd 72.09
B: mu 118.45 sd 57.21
corr: 0.02 P-value 0.795
100
200
300
400
bad
good
distOfNucl1ToPronLine

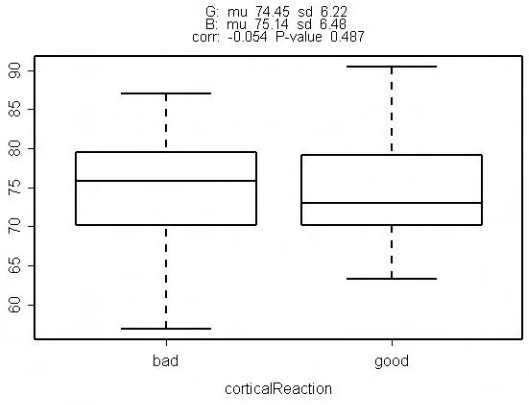

G: mu 74.15 sd 45.95
B: mu 85.04 sd 41.75
corr: -0.123 P-value 0.111
0
50
100
150
200
250
bad
good
distOfNucl2ToPronLine

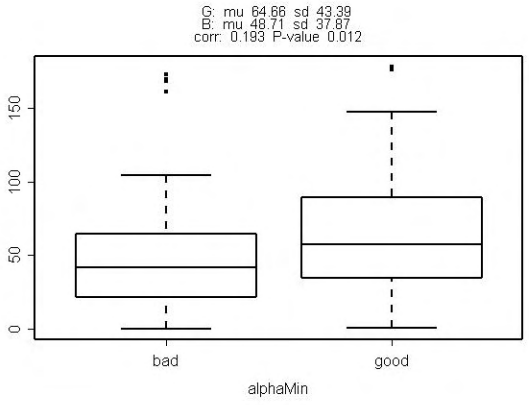

G: mu 59.31 sd 21.27
B: mu 67.48 sd 23.94
corr: -0.175 P-value 0.023
40
60
80
100
120
140
bad
good
distOfNucl1ToLineAvgRel

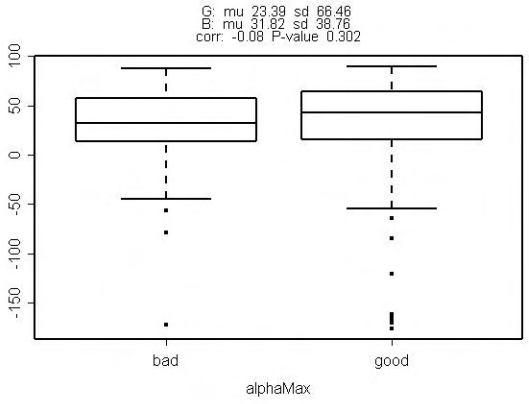

G: mu 59.74 sd 23.72
B: mu 66.06 sd 21.97
corr: -0.137 P-value 0.076
0
20
40
60
80
100
120
bad
good
distOfNucl2ToLineAvgRel

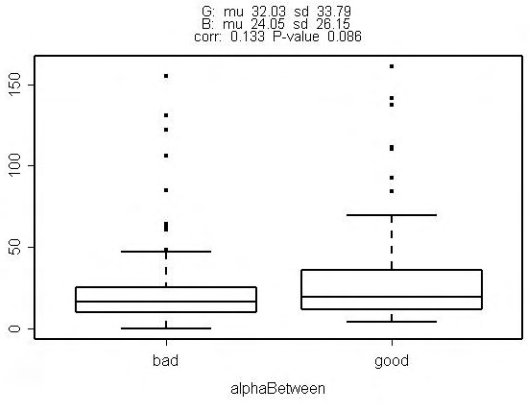

G: mu 9.18 sd 4
B: mu 8.59 sd 3.13
corr: 0.084 P-value 0.278
bad
good
lineVarOfNucl1Avg

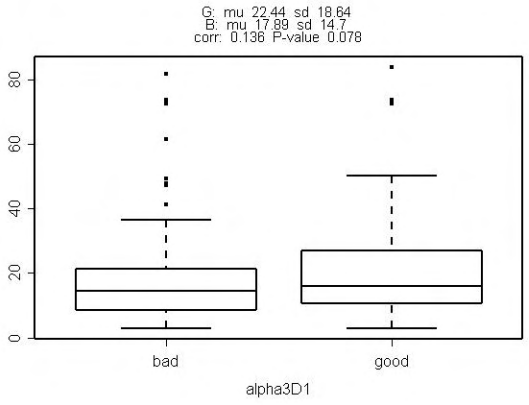

G: mu 6.73 sd 3.31
B: mu 7.86 sd 2.92
corr: -0.178 P-value 0.021
bad
good
lineVarOfNucl2Avg

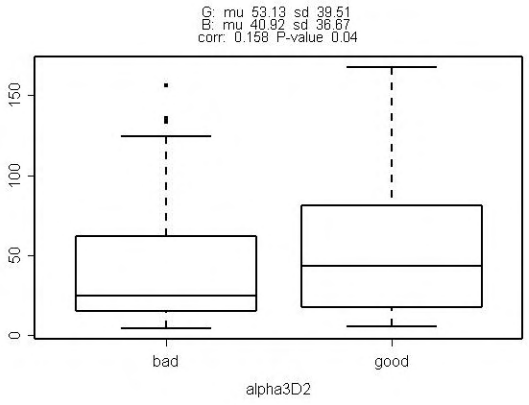

G: mu 74.45 sd 6.22
B: mu 75.14 sd 6.48
corr: -0.054 P-value 0.487
bad
good
corticalReaction

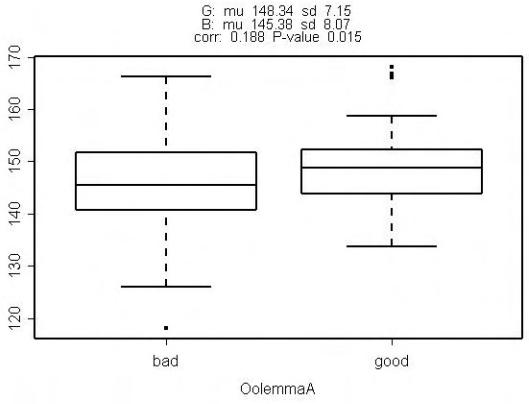

G: mu 64.66 sd 43.39
B: mu 48.71 sd 37.87
corr: 0.193 P-value 0.012
bad
good
alphaMin

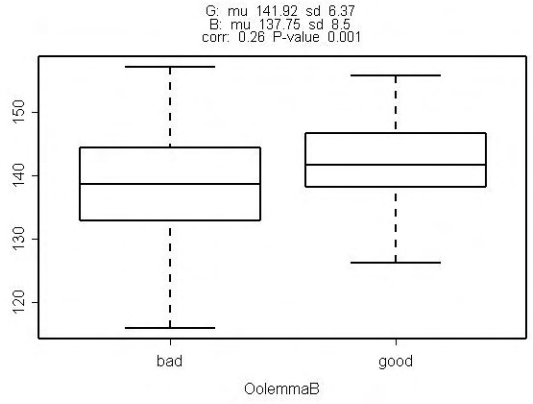

G: mu 23.39 sd 66.46
B: mu 31.82 sd 38.76
corr: -0.08 P-value 0.302
bad
good
alphaMax

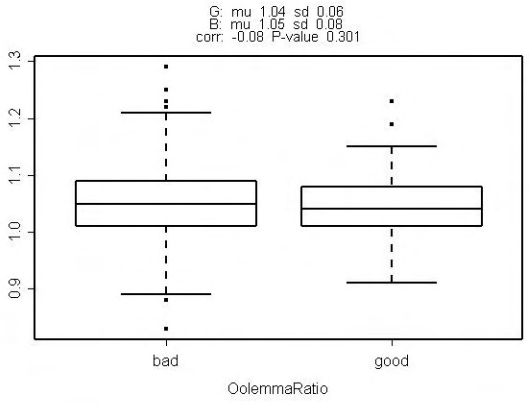

G: mu 32.03 sd 33.79
B: mu 24.05 sd 26.15
corr: 0.133 P-value 0.086
bad
good
alphaBetween

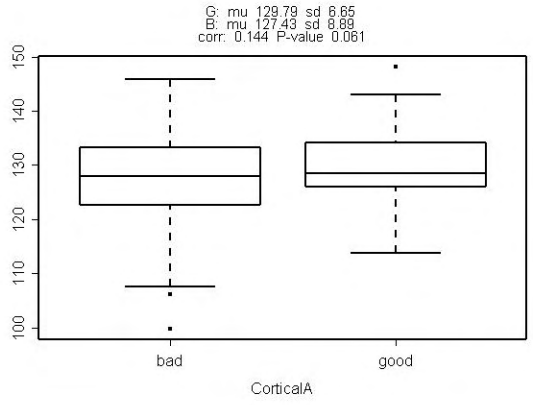

G: mu 22.44 sd 18.64
B: mu 17.88 sd 14.7
corr: 0.136 P-value 0.078
bad
good
alpha3D1

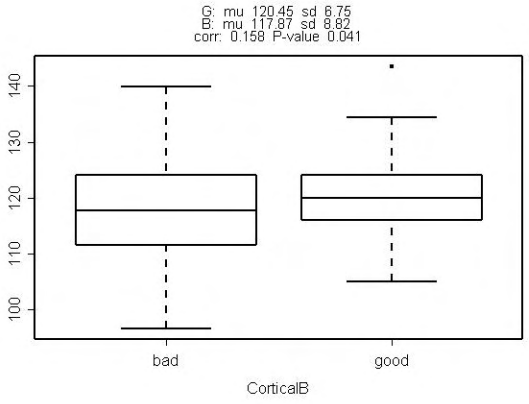

G: mu 53.13 sd 39.51
B: mu 40.92 sd 36.67
corr: 0.158 P-value 0.04
bad
good
alpha3D2

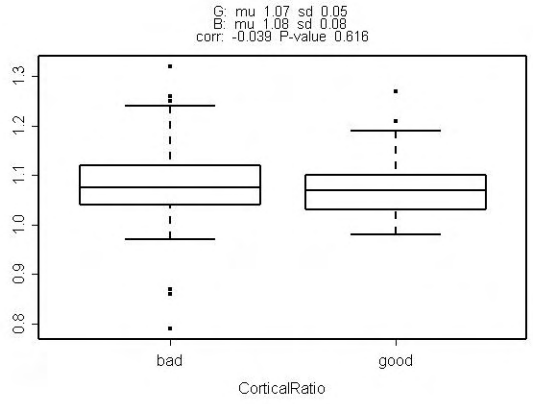

G: mu 148.34 sd 7.15
B: mu 145.38 sd 8.07
corr: 0.188 P-value 0.015
bad
good
OolemmaA

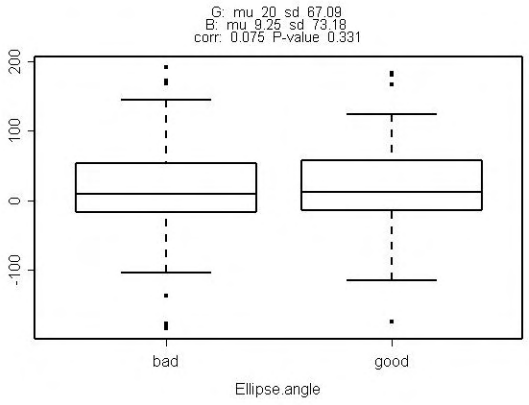

G: mu 141.92 sd 6.37
B: mu 137.75 sd 8.5
corr: 0.26 P-value 0.001
bad
good
OolemmaB

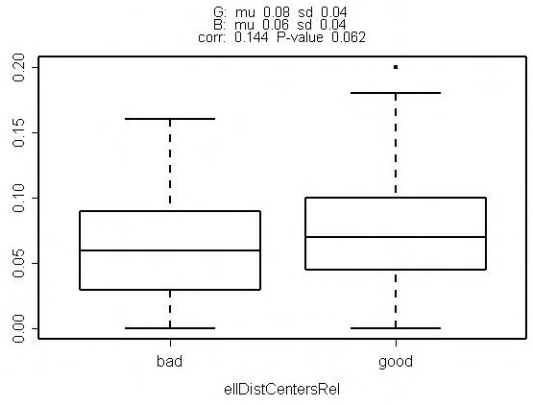

G: mu 1.04 sd 0.06
B: mu 1.05 sd 0.08
corr: -0.08 P-value 0.301
bad
good
OolemmaRatio

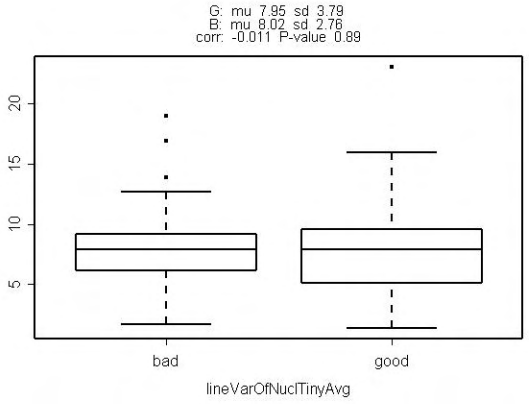

G: mu 129.79 sd 6.65
B: mu 127.43 sd 8.89
corr: 0.144 P-value 0.061
bad
good
CorticalA

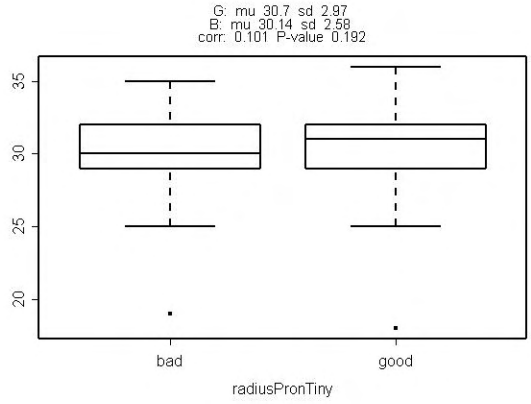

G: mu 120.45 sd 6.75
B: mu 117.87 sd 8.82
corr: 0.158 P-value 0.041
bad
good
CorticalB

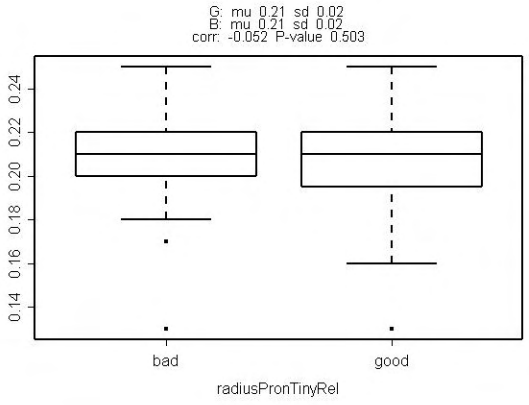

G: mu 1.07 sd 0.05
B: mu 1.08 sd 0.08
corr: -0.039 P-value 0.616
bad
good
CorticalRatio

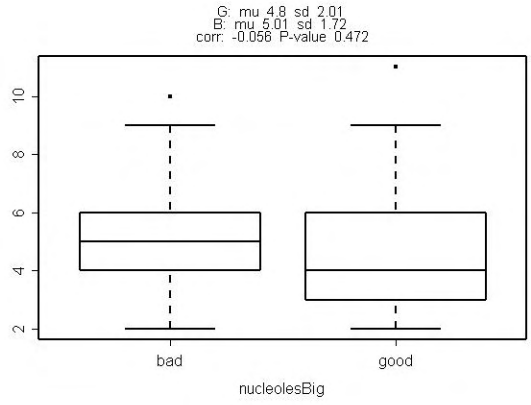

G: mu 20 sd 67.09
B: mu 9.25 sd 73.18
corr: 0.075 P-value 0.331
bad
good
Ellipse.angle

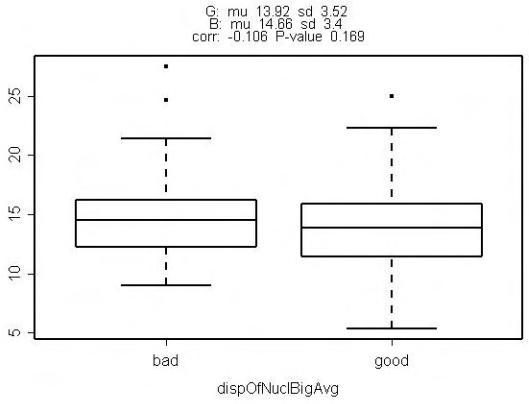

G: mu 0.08 sd 0.04
B: mu 0.06 sd 0.04
corr: 0.144 P-value 0.062
bad
good
ellDistCentersRel

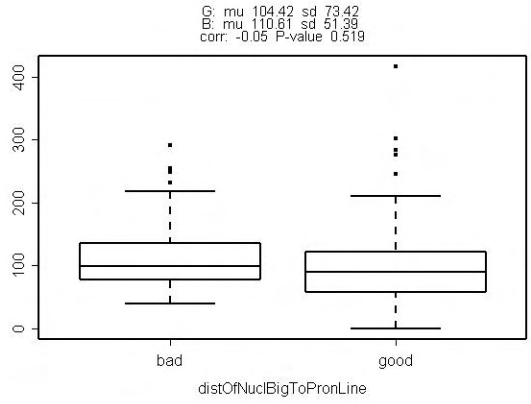

G: mu 7.95 sd 3.79
B: mu 8.02 sd 2.76
corr: -0.011 P-value 0.89
5
10
15
20
bad
good
lineVarOfNuclTinyAvg

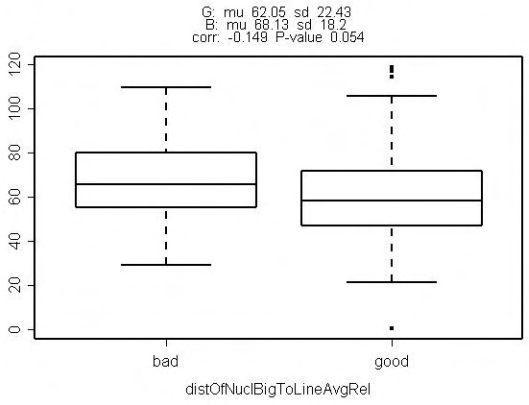

G: mu 30.7 sd 2.97
B: mu 30.14 sd 2.58
corr: 0.101 P-value 0.192
20
25
30
35
bad
good
radiusPronTiny

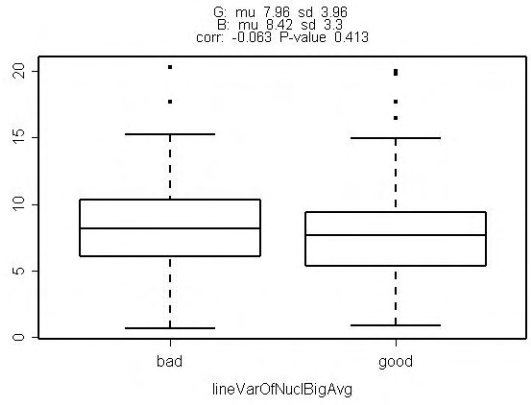

G: mu 0.21 sd 0.02
B: mu 0.21 sd 0.02
corr: -0.052 P-value 0.503
0.14
0.16
0.18
0.20
0.22
0.24
bad
good
radiusPronTinyRel

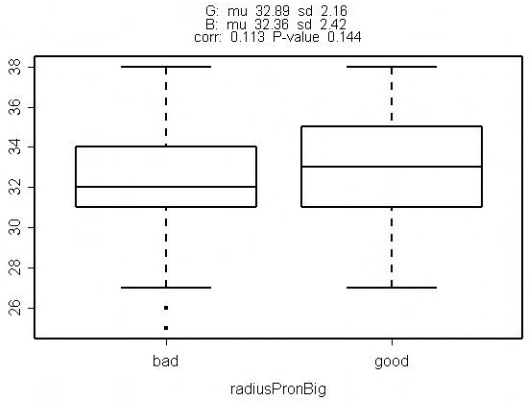

G: mu 4.8 sd 2.01
B: mu 5.01 sd 1.72
corr: -0.056 P-value 0.472
2
4
6
8
10
bad
good
nucleolesBig

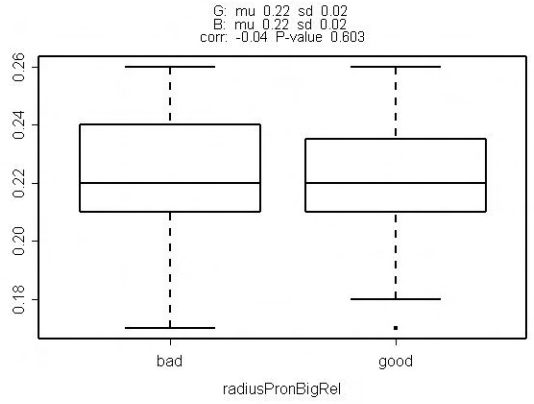

G: mu 13.92 sd 3.52
B: mu 14.66 sd 3.4
corr: -0.106 P-value 0.169
5
10
15
20
25
bad
good
dispOfNuclBigAvg

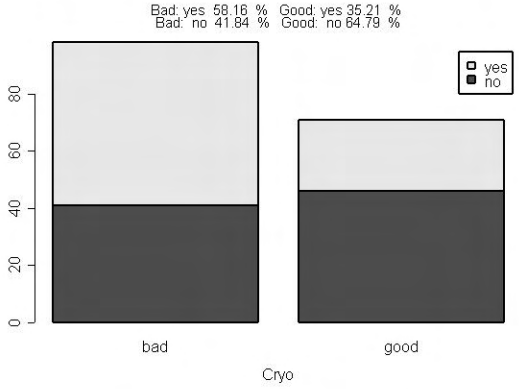

G: mu 104.42 sd 73.42
B: mu 110.61 sd 51.39
corr: -0.05 P-value 0.519
0
100
200
300
400
bad
good
distOfNuclBigToPronLine


G: mu 62.05 sd 22.43
B: mu 68.13 sd 18.2
corr: -0.149 P-value 0.054
0
20
40
60
80
100
120
bad
good
distOfNuclBigToLineAvgRel


G: mu 7.96 sd 3.96
B: mu 8.42 sd 3.3
corr: -0.063 P-value 0.413
0
5
10
15
20
bad
good
lineVarOfNuclBigAvg


G: mu 32.89 sd 2.16
B: mu 32.36 sd 2.42
corr: 0.113 P-value 0.144
26
28
30
32
34
36
38
bad
good
radiusPronBig

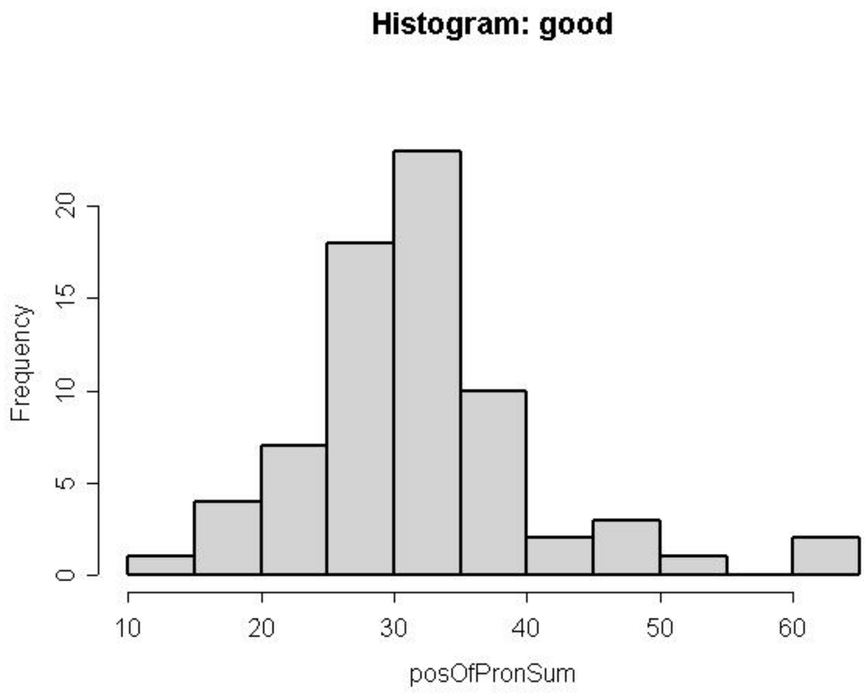

G: mu 0.22 sd 0.02
B: mu 0.22 sd 0.02
corr: -0.04 P-value 0.603
0.18
0.20
0.22
0.24
0.26
bad
good
radiusPronBigRel

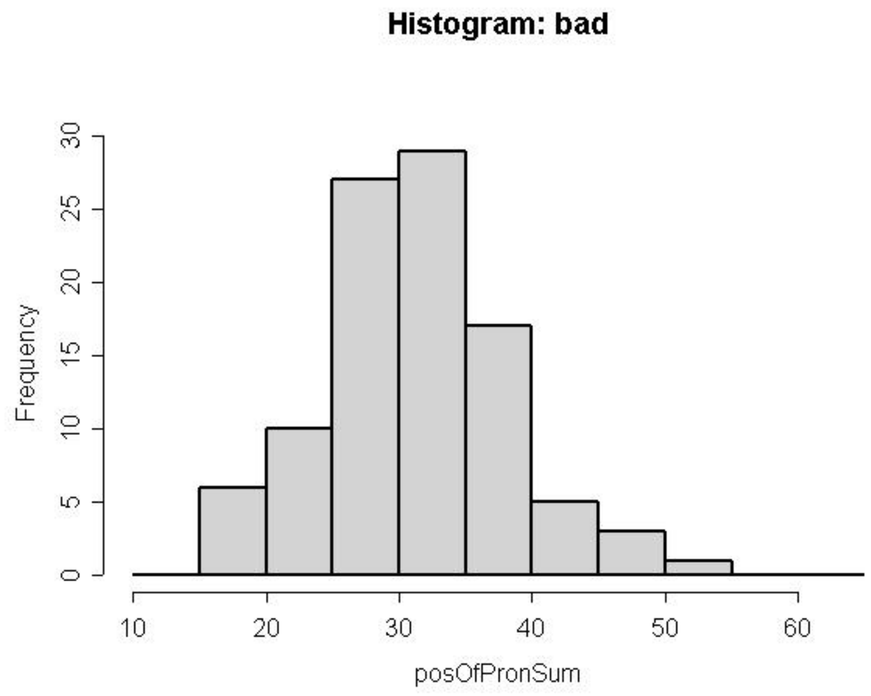

Bad: yes 58.16 % Good: yes 35.21 %
Bad: no 41.84 % Good: no 64.79 %
yes
no
0
20
40
60
80
bad
good
Cryo

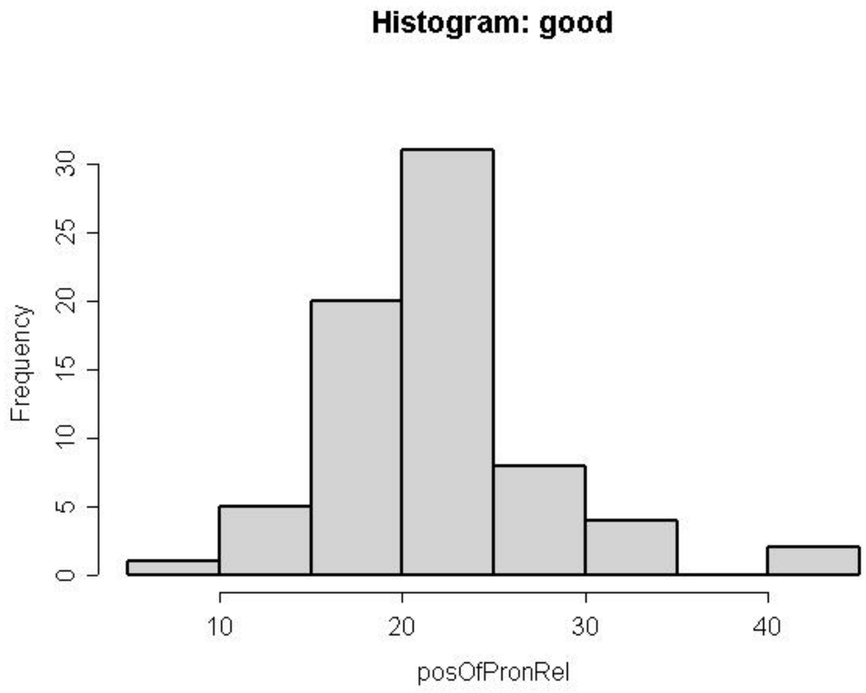
Appendix B

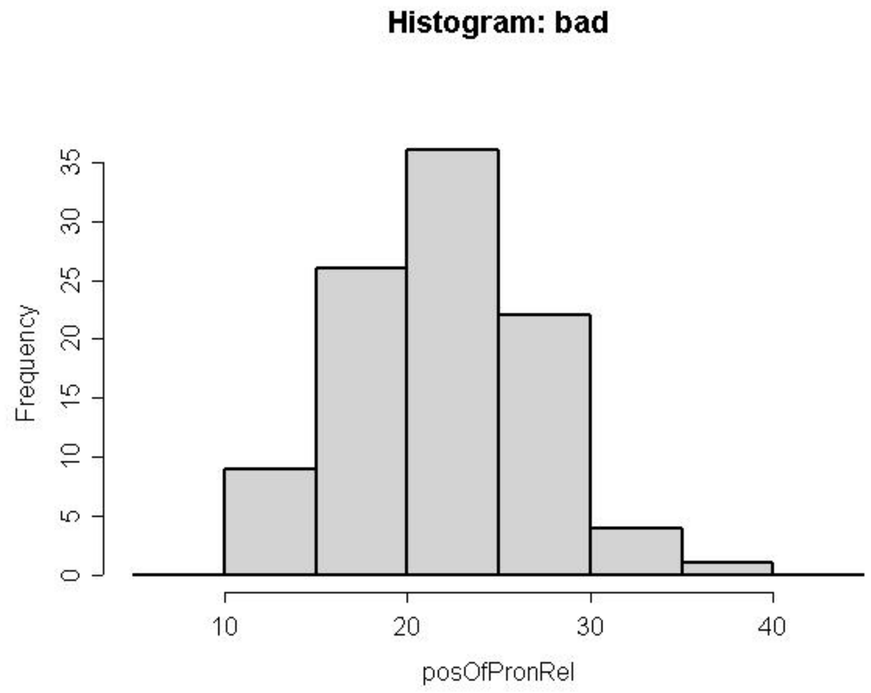
Histograms for each class

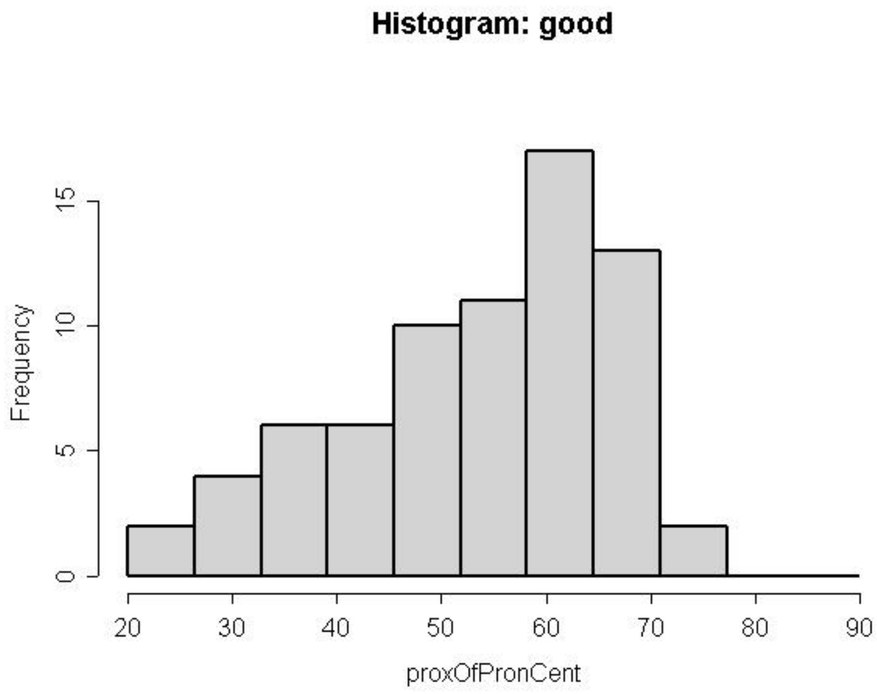
advantages:	needs no explanation, familiar
disadvantages:	contain no information on distribution within the bins, sensitive to number and placement of bins

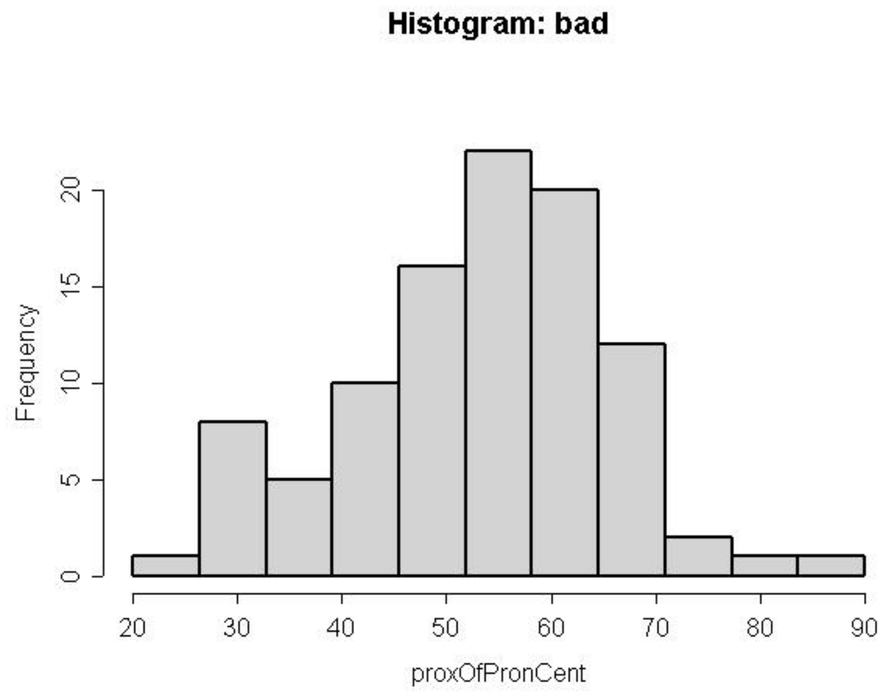

Histogram: good
Frequency
0
5
10
15
20
10
20
30
40
50
60
posOfPronSum

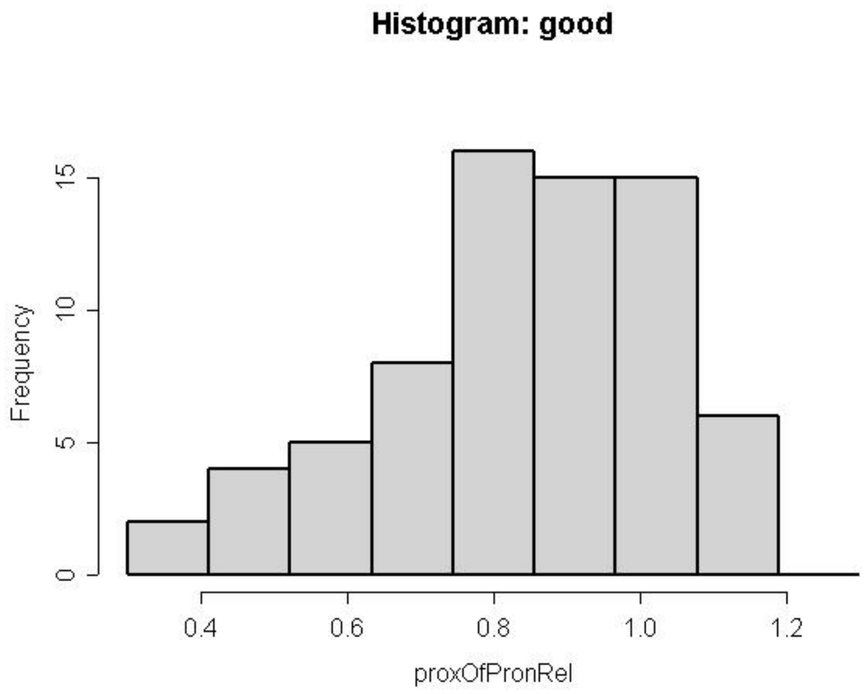

Histogram: bad
Frequency
0
5
10
15
20
25
30
10
20
30
40
50
60
posOfPronSum

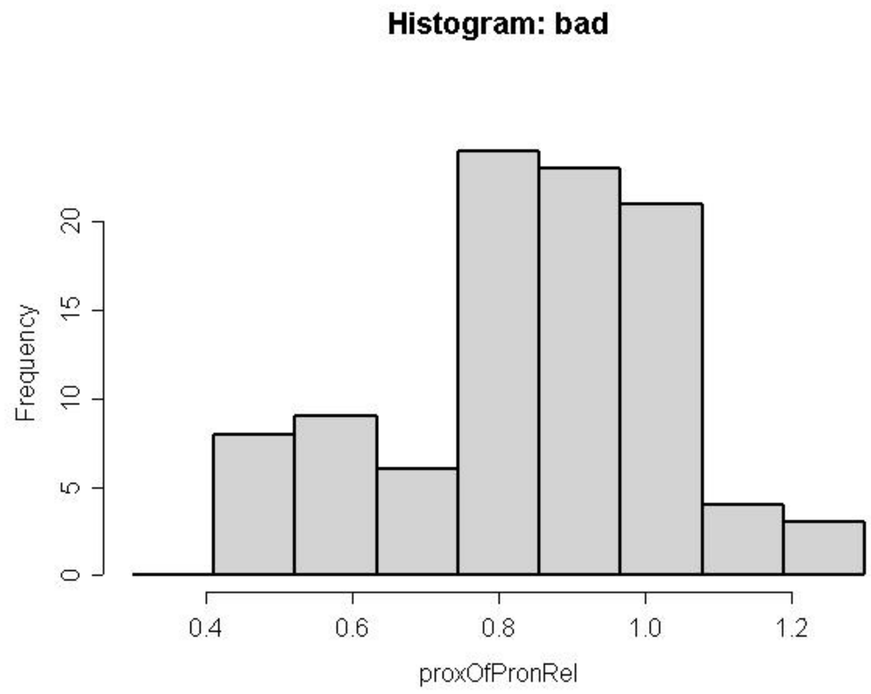

Histogram: good
Frequency
0
5
10
15
20
25
30
10
20
30
40
posOfPronRel

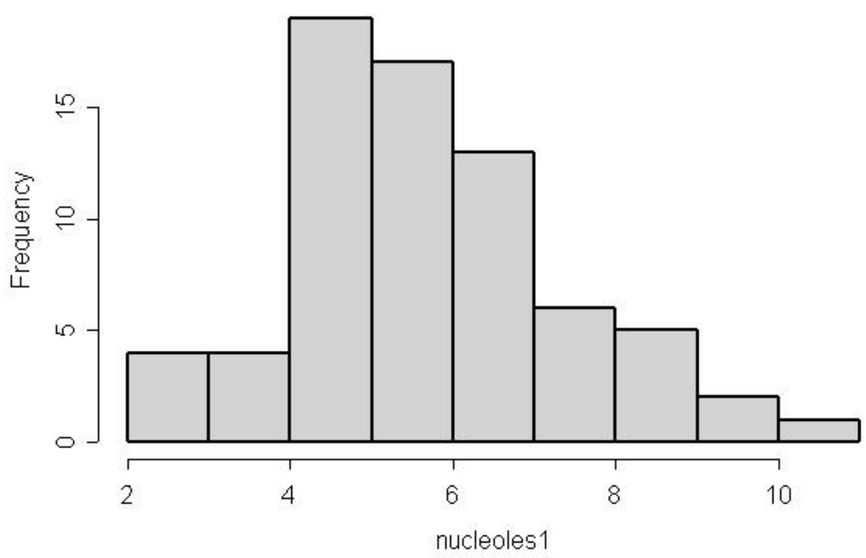

Histogram: bad
Frequency
0
5
10
15
20
25
30
35
10
20
30
40
posOfPronRel

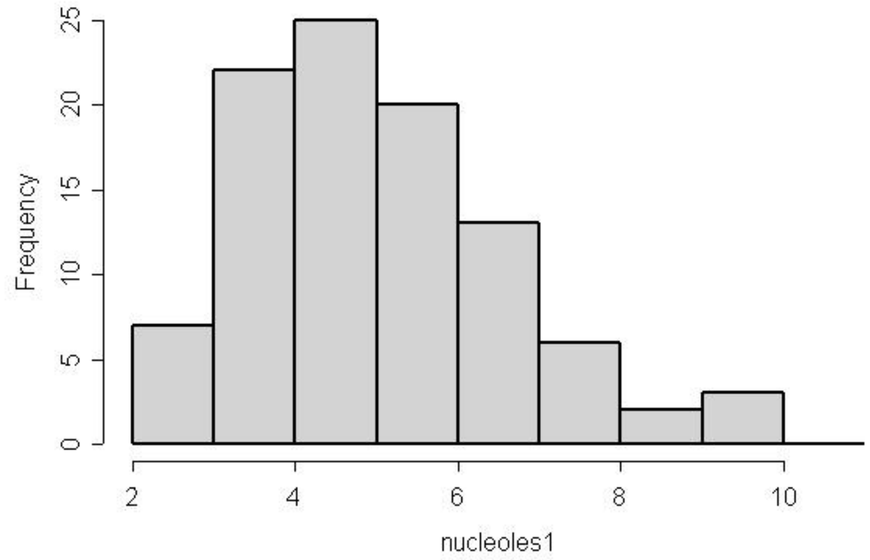

Histogram: good
Frequency
0
5
10
15
20
30
40
50
60
70
80
90
proxOfPronCent

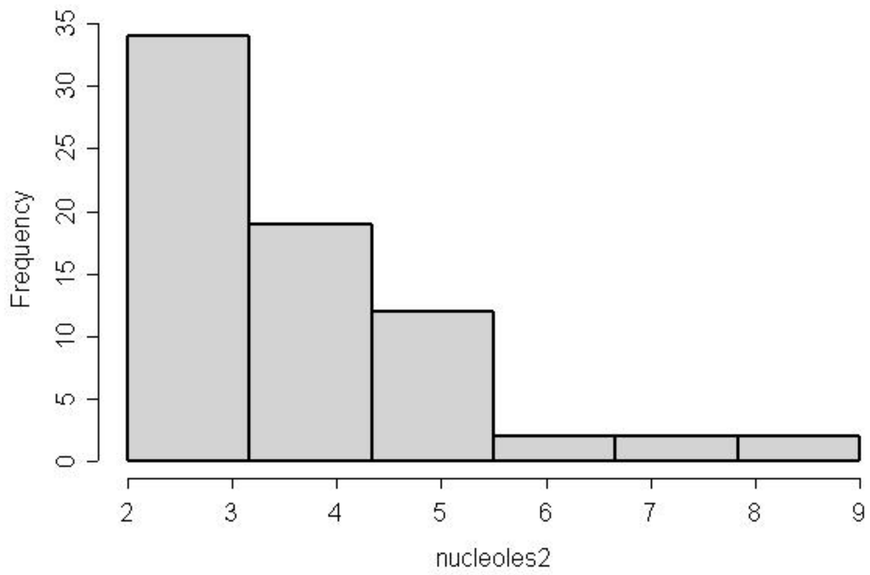

Histogram: bad
Frequency
0
5
10
15
20
20
30
40
50
60
70
80
90
proxOfPronCent

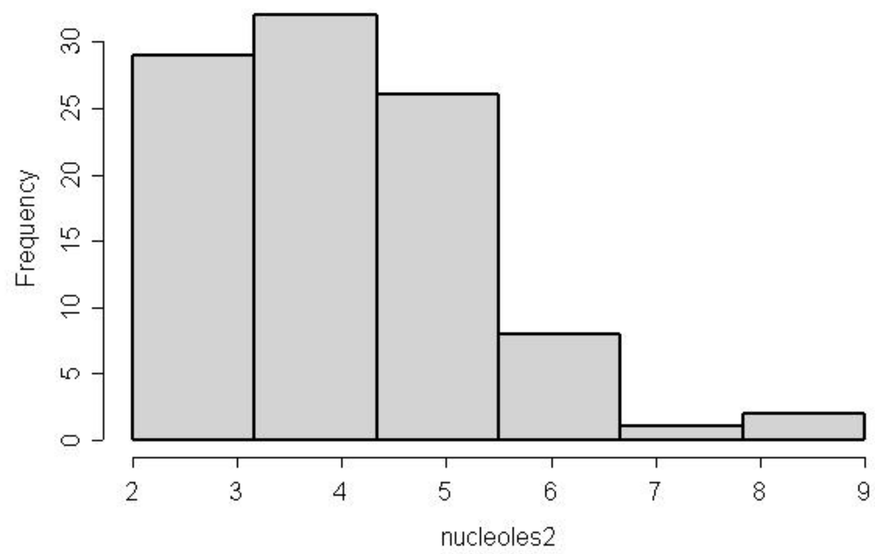

Histogram: good
Frequency
0
5
10
15
0.4
0.6
0.8
1.0
1.2
proxOfPronRel

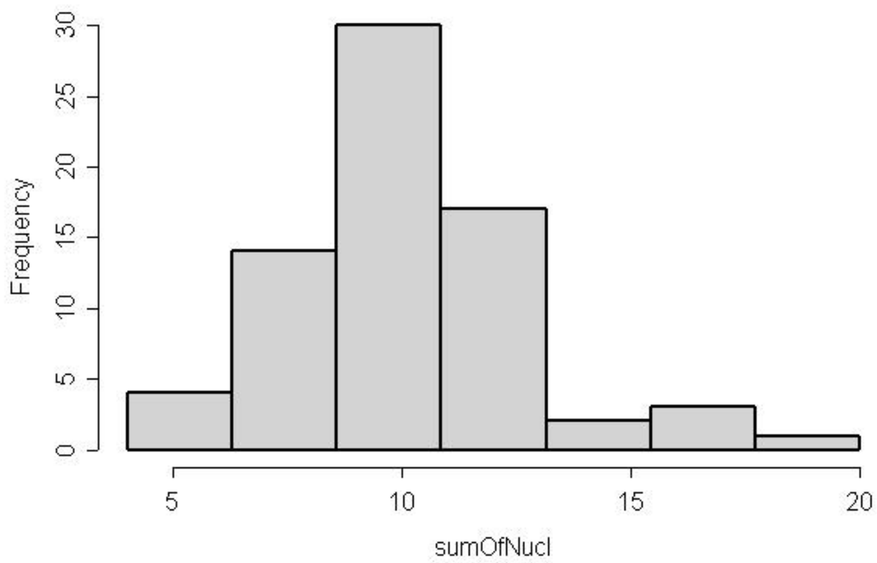

Histogram: bad
Frequency
0
5
10
15
20
0.4
0.6
0.8
1.0
1.2
proxOfPronRel

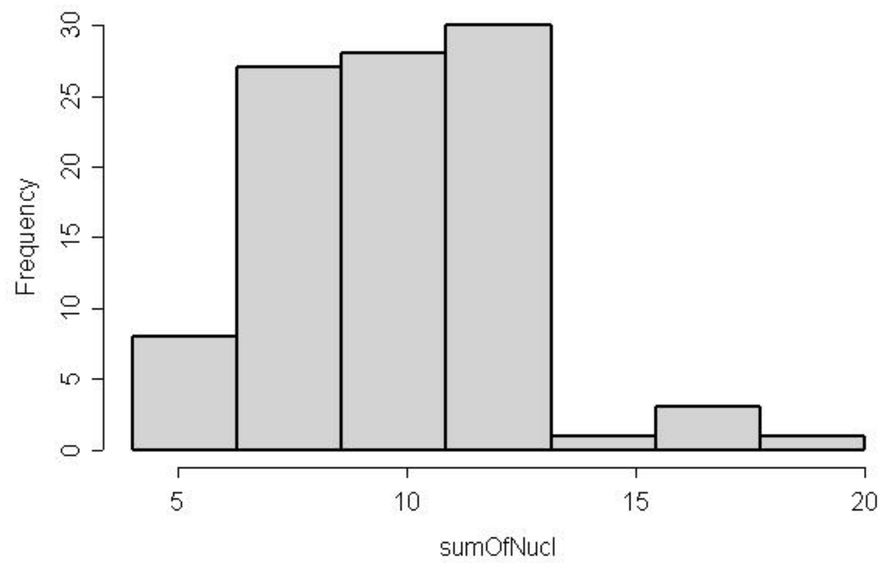

Histogram: good
Frequency
0
5
10
15
2
4
6
8
10
nucleoles1

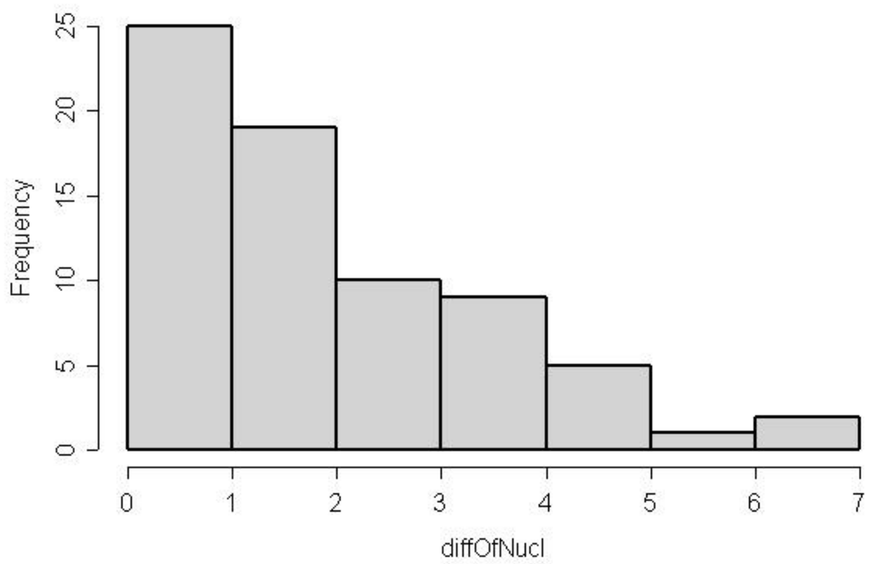

Histogram: bad
Frequency
0
5
10
15
20
25
2
4
6
8
10
nucleoles1

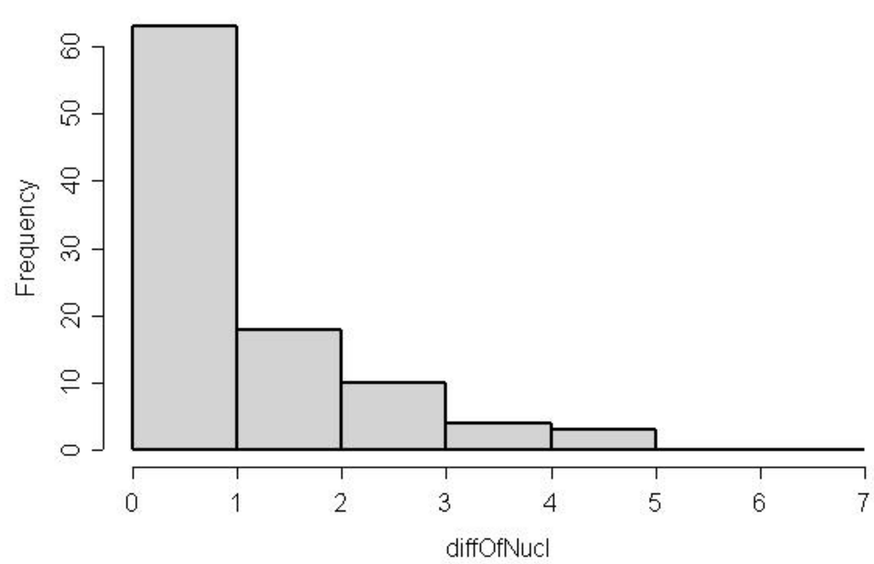

Histogram: good
Frequency
0
5
10
15
20
25
30
35
2
3
4
5
6
7
8
9
nucleoles2

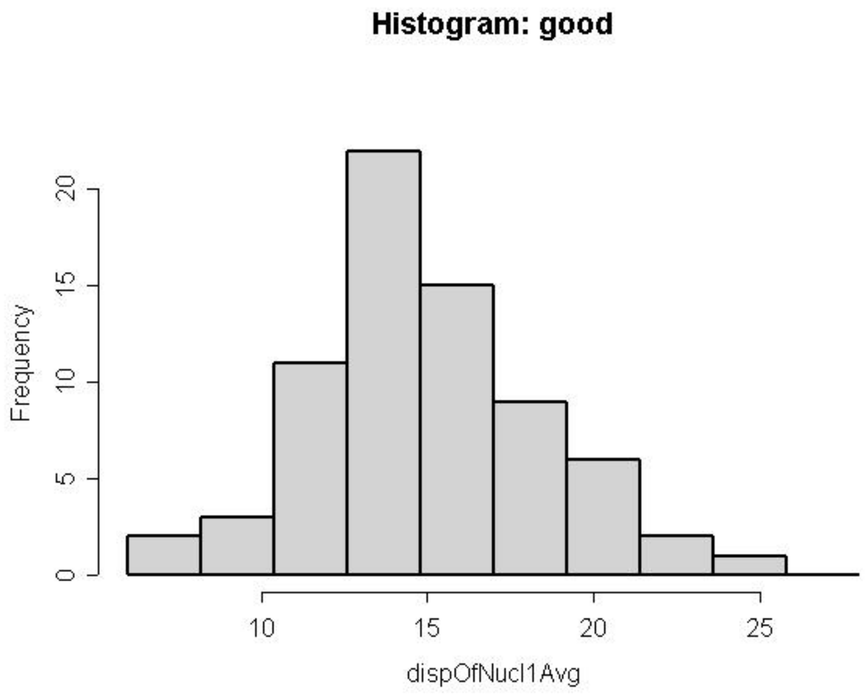

Histogram: bad
Frequency
0
5
10
15
20
25
30
2
3
4
5
6
7
8
9
nucleoles2

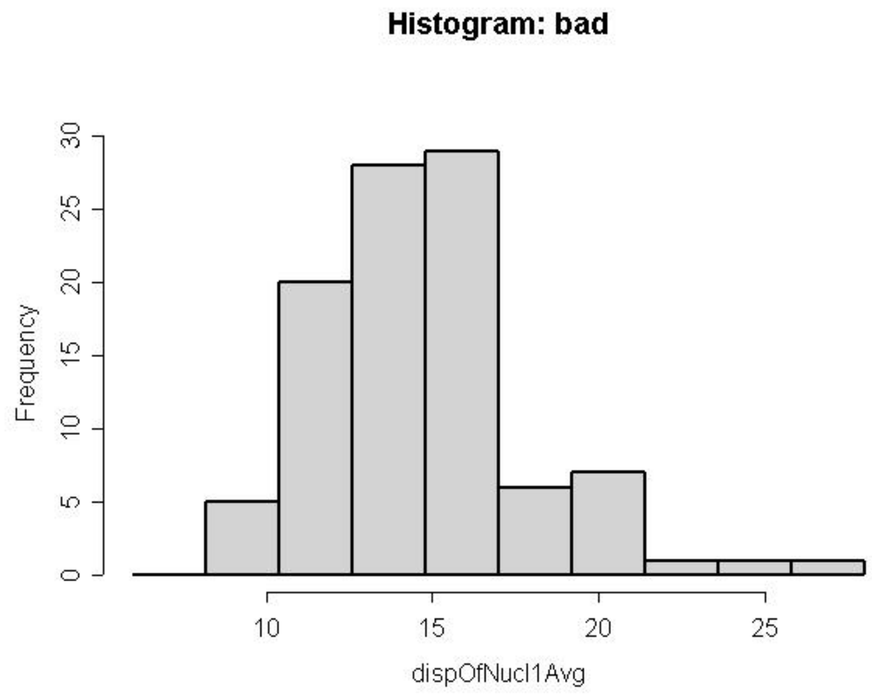

Histogram: good
Frequency
0
5
10
15
20
25
30
5
10
15
20
sumOfNucl

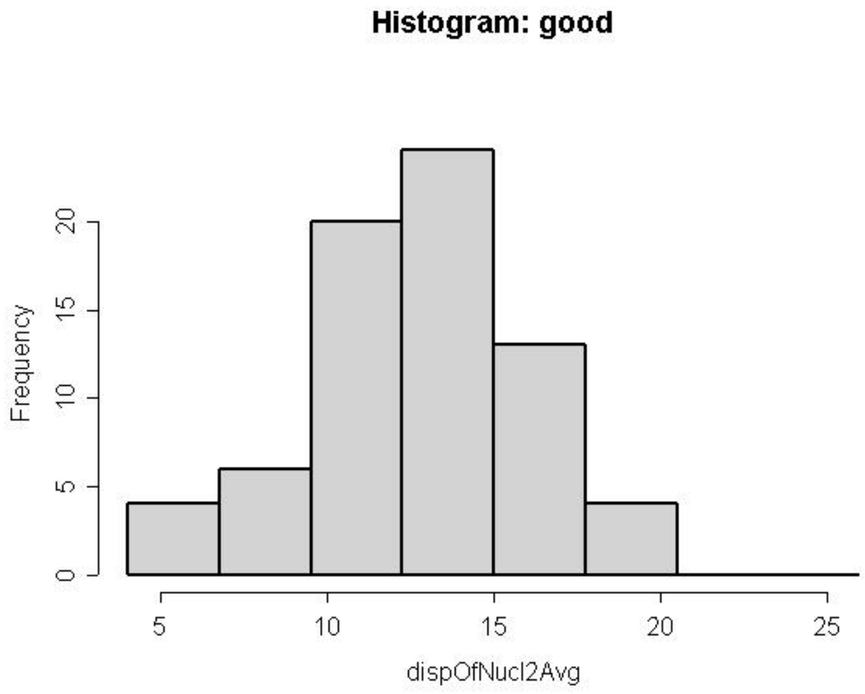

Histogram: bad
Frequency
0
5
10
15
20
25
30
5
10
15
20
sumOfNucl

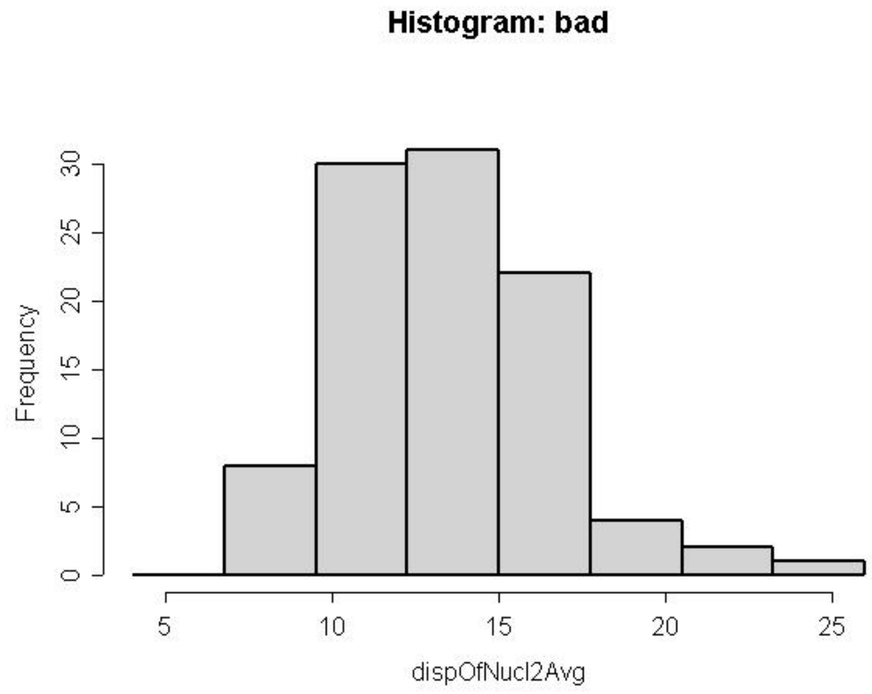

Histogram: good
Frequency
0
5
10
15
20
25
0
1
2
3
4
5
6
7
diffOfNucl

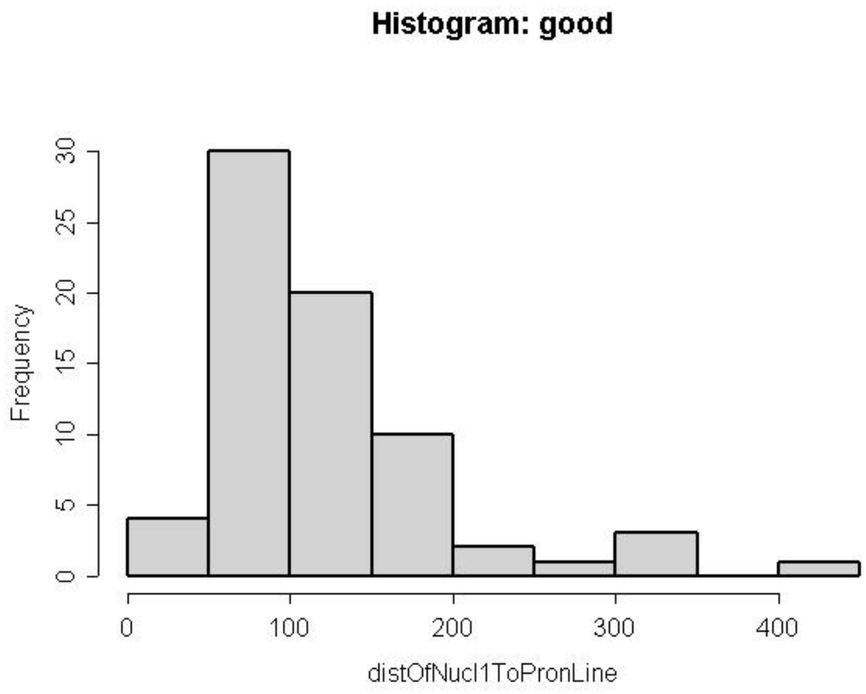

Histogram: bad
Frequency
0
10
20
30
40
50
60
0
1
2
3
4
5
6
7
diffOfNucl

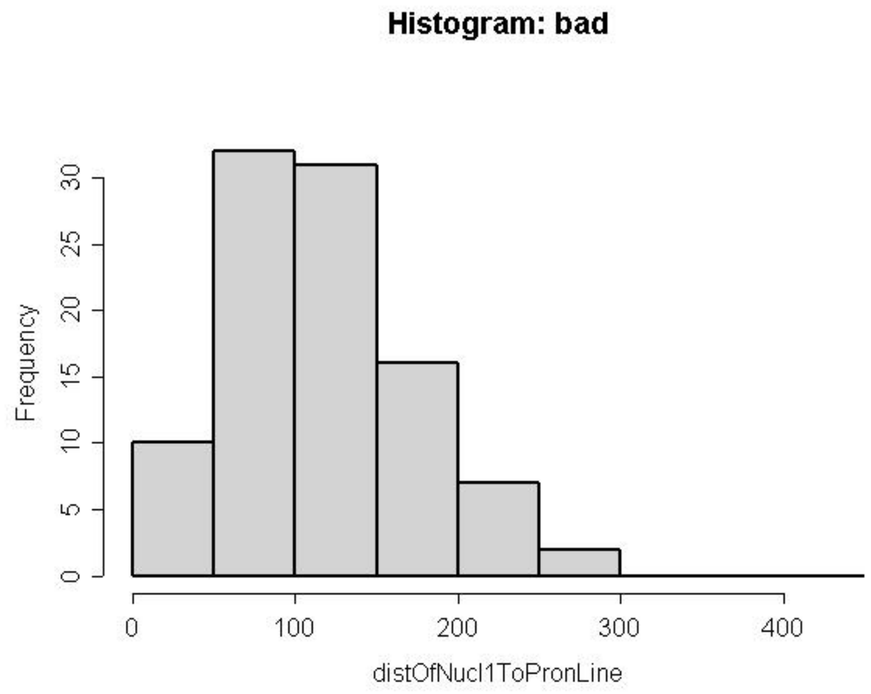

Histogram: good
Frequency
0
5
10
15
20
10
15
20
25
dispOfNucl1Avg

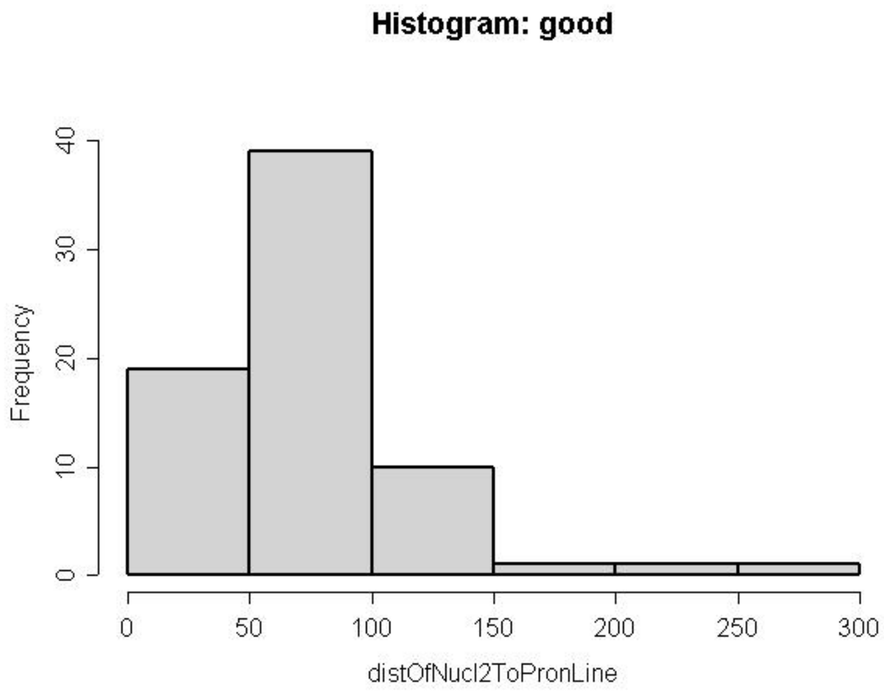

Histogram: bad
Frequency
0
5
10
15
20
25
30
10
15
20
25
dispOfNucl1Avg

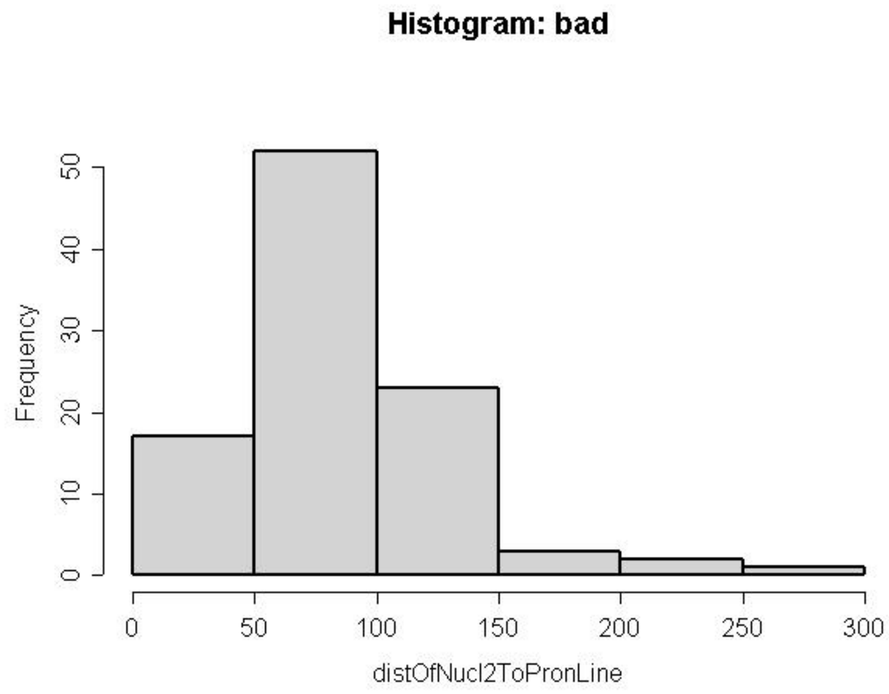

Histogram: good
Frequency
0
5
10
15
20
5
10
15
20
25
dispOfNucl2Avg

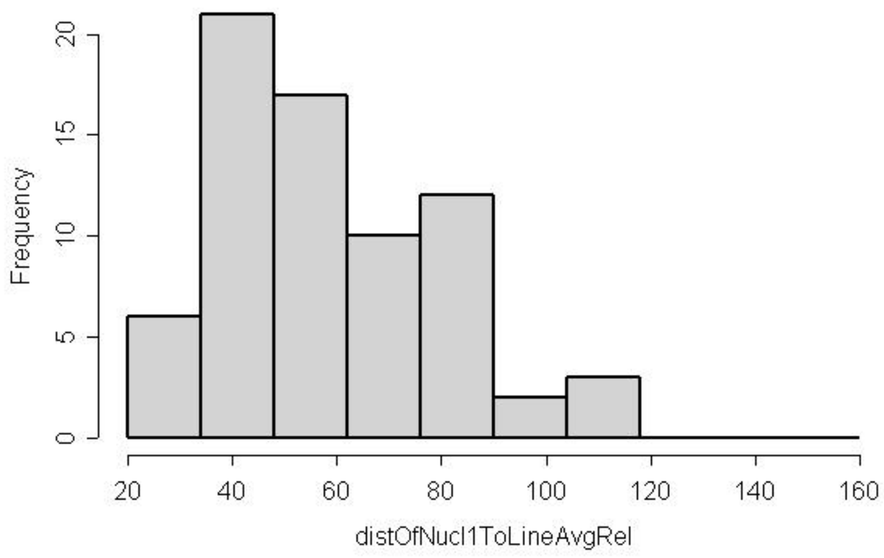

Histogram: bad
Frequency
0
5
10
15
20
25
30
5
10
15
20
25
dispOfNucl2Avg

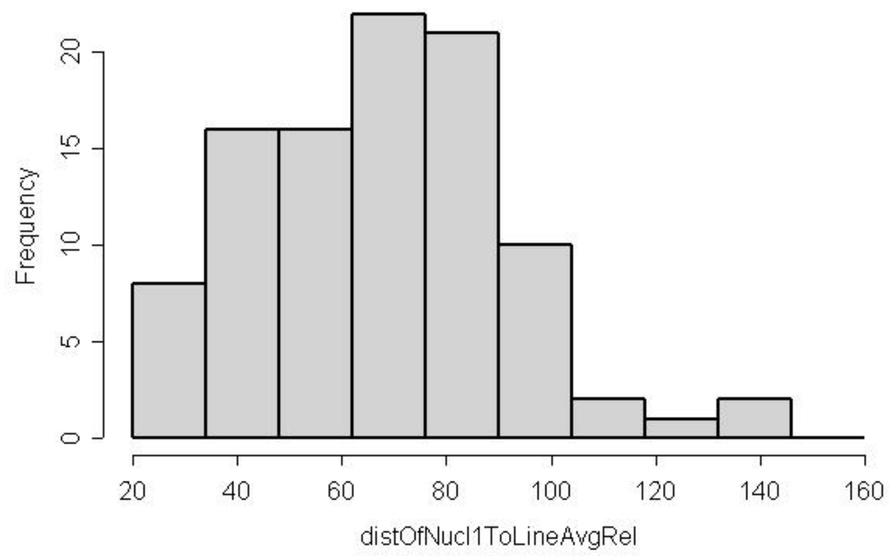

Histogram: good
Frequency
0
5
10
15
20
25
30
0
100
200
300
400
distOfNucl1ToPronLine

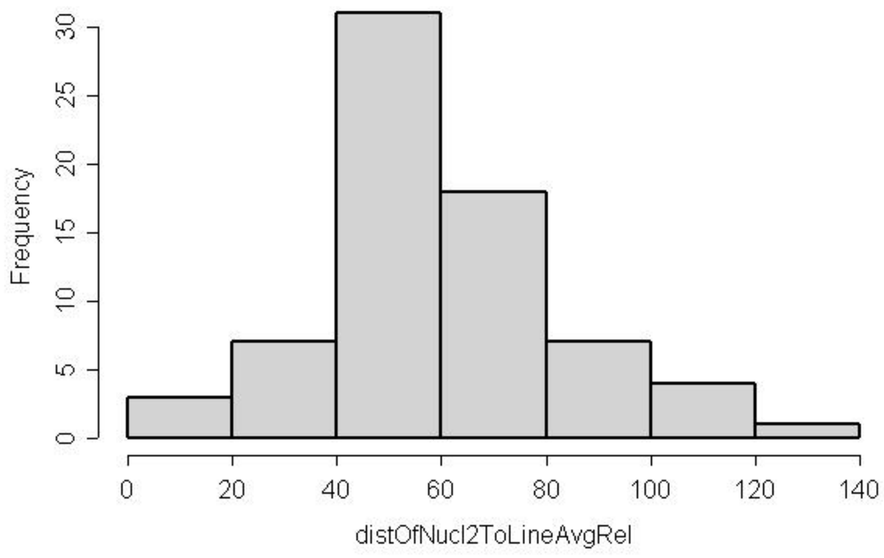

Histogram: bad
Frequency
0
5
10
15
20
25
30
0
100
200
300
400
distOfNucl1ToPronLine

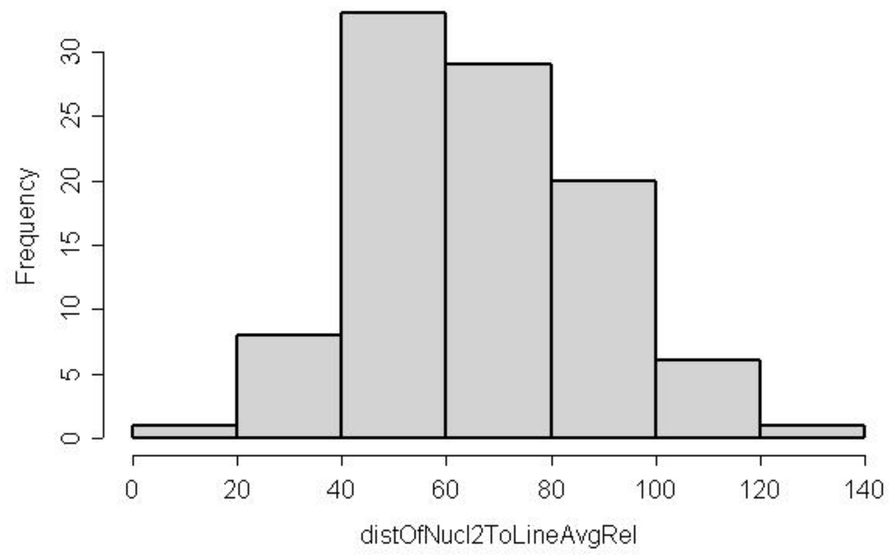

Histogram: good
Frequency
0
10
20
30
40
0
50
100
150
200
250
300
distOfNucl2ToPronLine

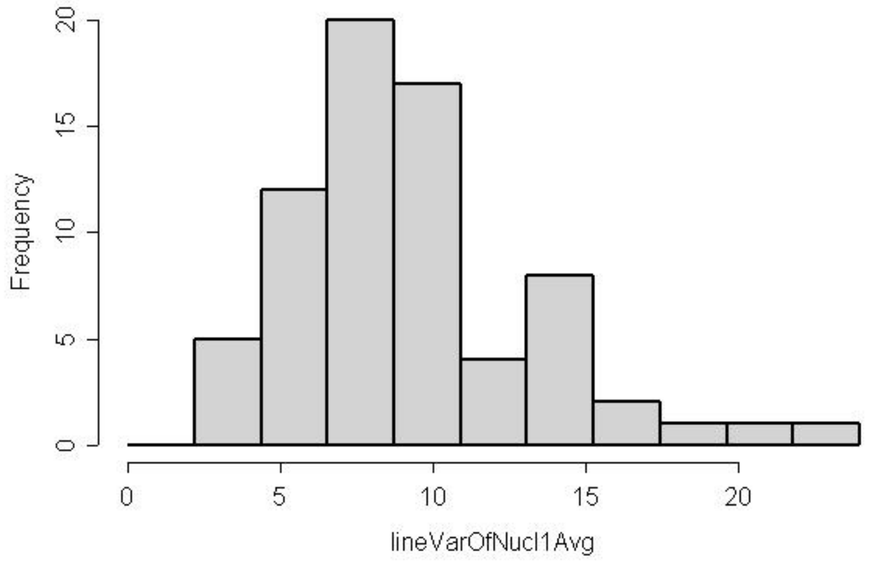

Histogram: bad
Frequency
0
10
20
30
40
50
0
50
100
150
200
250
300
distOfNucl2ToPronLine

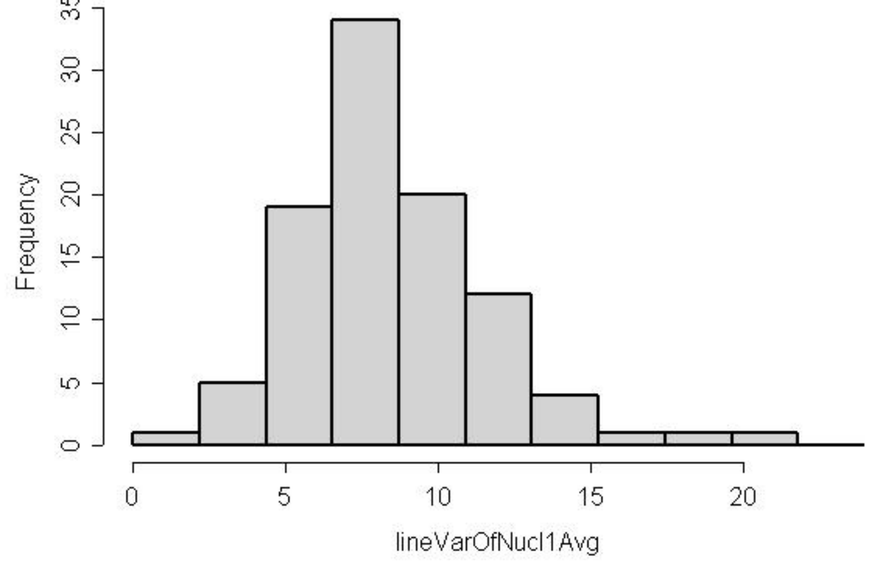

Histogram: good
Frequency
0
5
10
15
20
20
40
60
80
100
120
140
160
distOfNucl1ToLineAvgRel

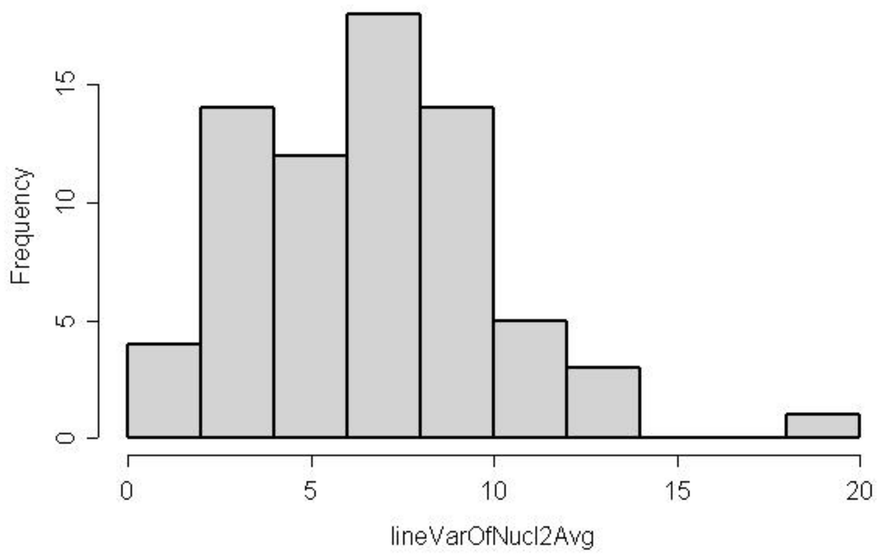

Histogram: bad
Frequency
0
5
10
15
20
20
40
60
80
100
120
140
160
distOfNucl1ToLineAvgRel

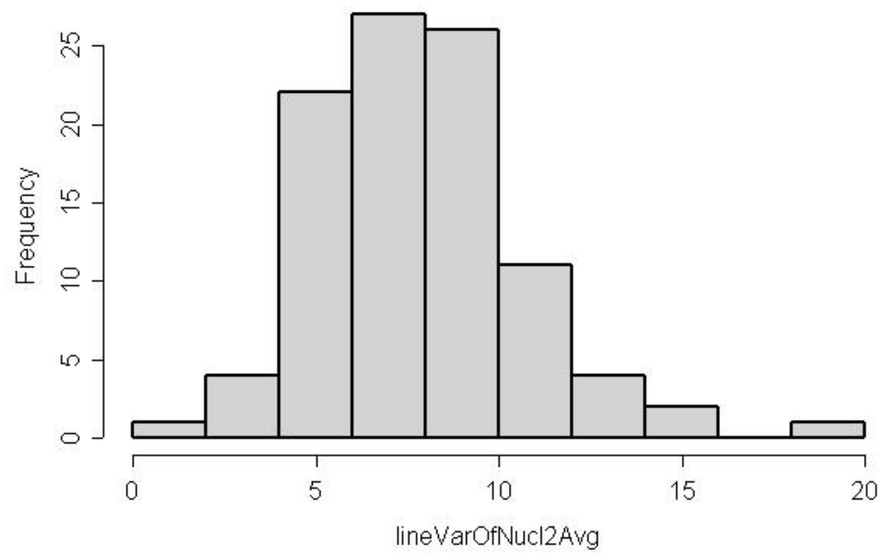

Histogram: good
Frequency
0
5
10
15
20
25
30
0
20
40
60
80
100
120
140
distOfNucl2ToLineAvgRel

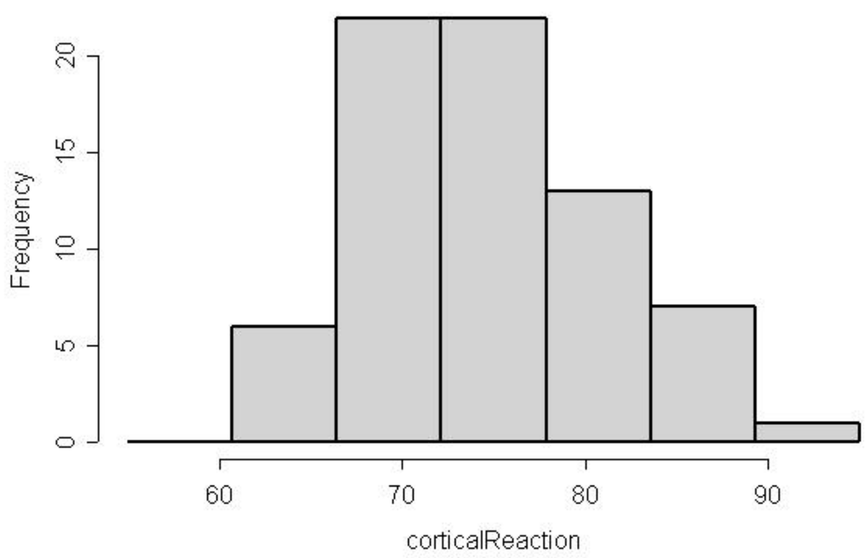

Histogram: bad
Frequency
0
5
10
15
20
25
30
0
20
40
60
80
100
120
140
distOfNucl2ToLineAvgRel

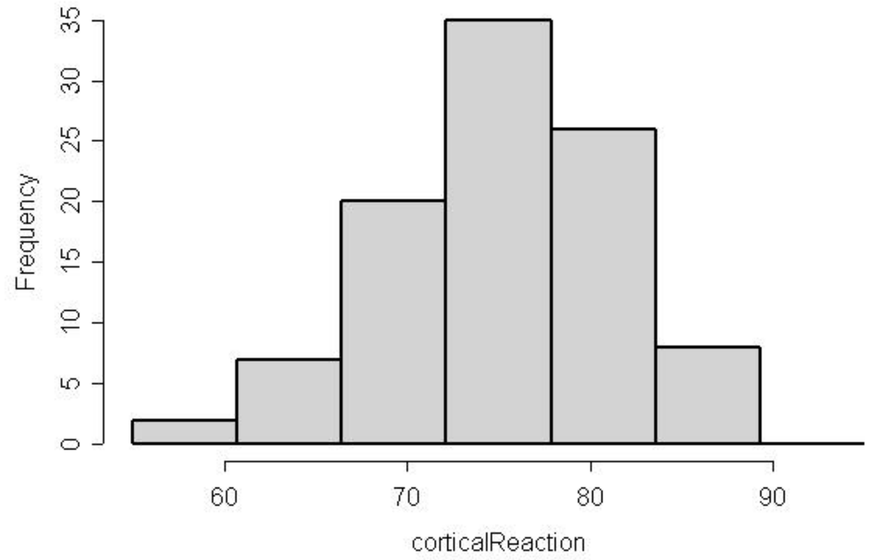

Histogram: good
Frequency
0
5
10
15
20
0
5
10
15
20
lineVarOfNucl1Avg

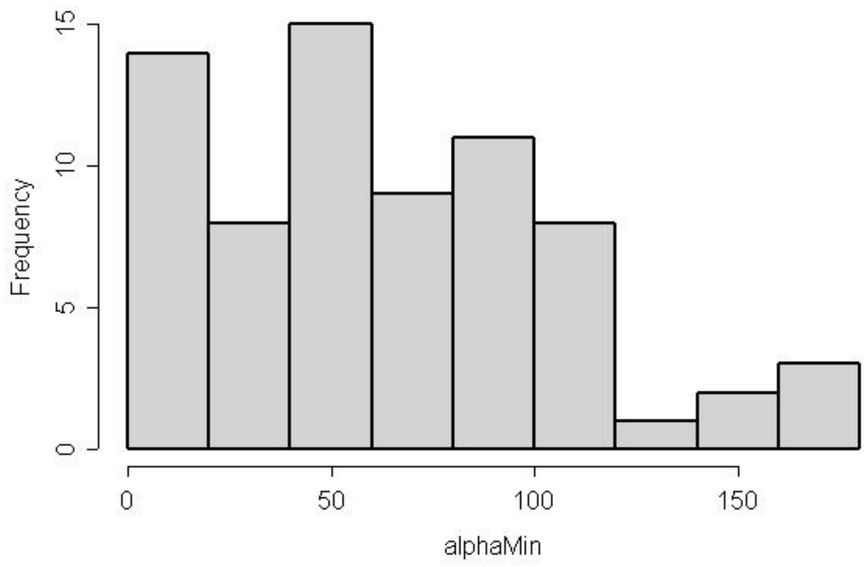

Histogram: bad
Frequency
0
5
10
15
20
25
30
35
0
5
10
15
20
lineVarOfNucl1Avg

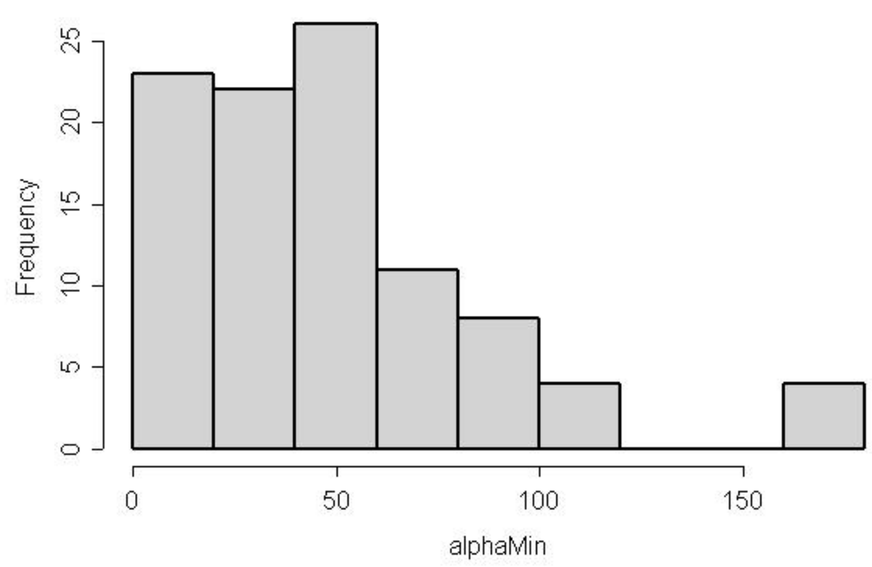

Histogram: good
Frequency
0
5
10
15
0
5
10
15
20
lineVarOfNucl2Avg

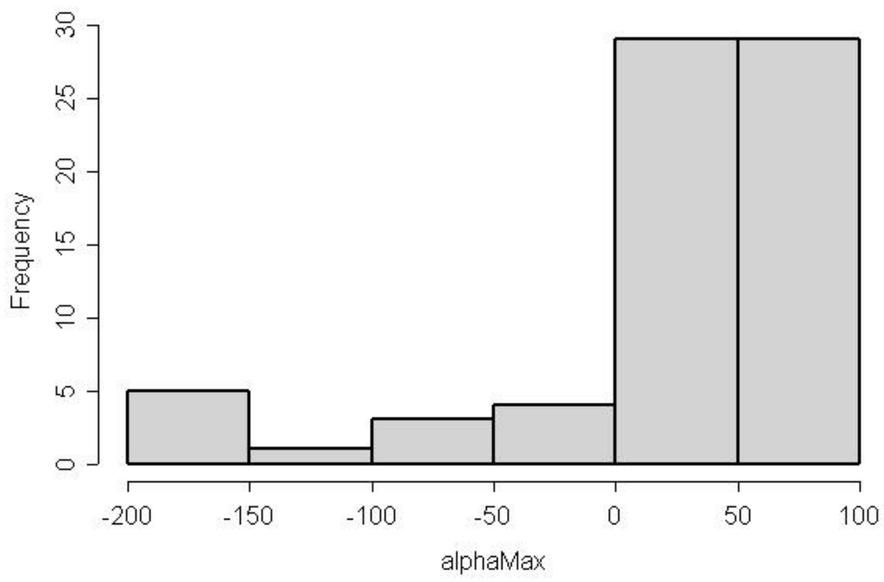

Histogram: bad
Frequency
0
5
10
15
20
25
0
5
10
15
20
lineVarOfNucl2Avg

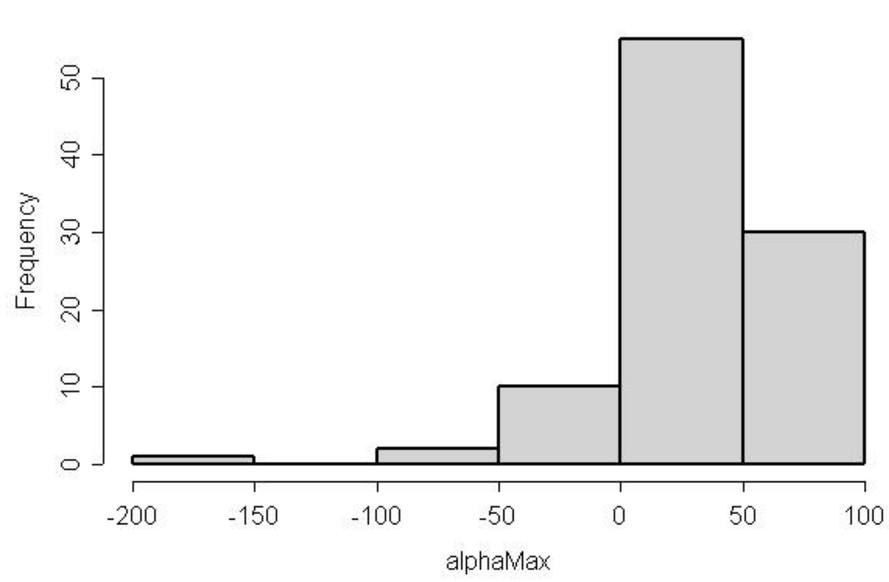

Histogram: good
Frequency
0
5
10
15
20
60
70
80
90
corticalReaction

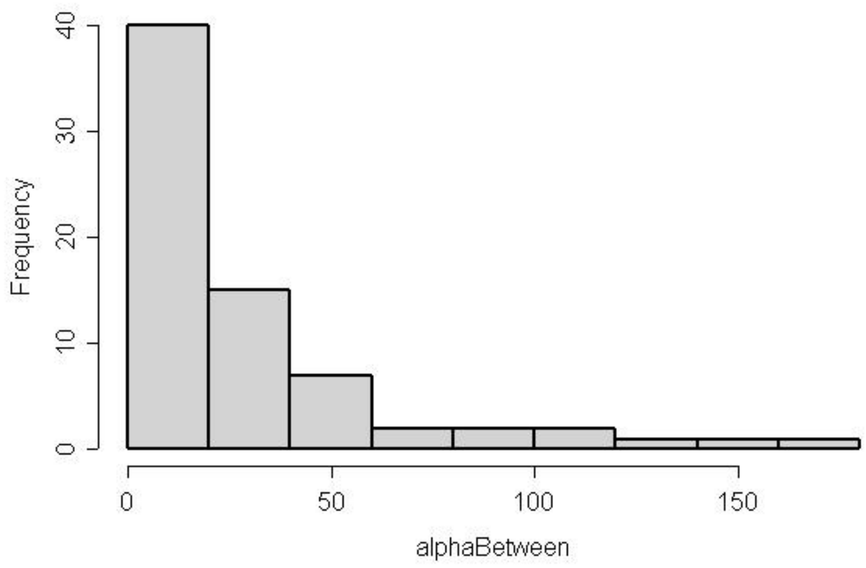

Histogram: bad
Frequency
0
5
10
15
20
25
30
35
60
70
80
90
corticalReaction

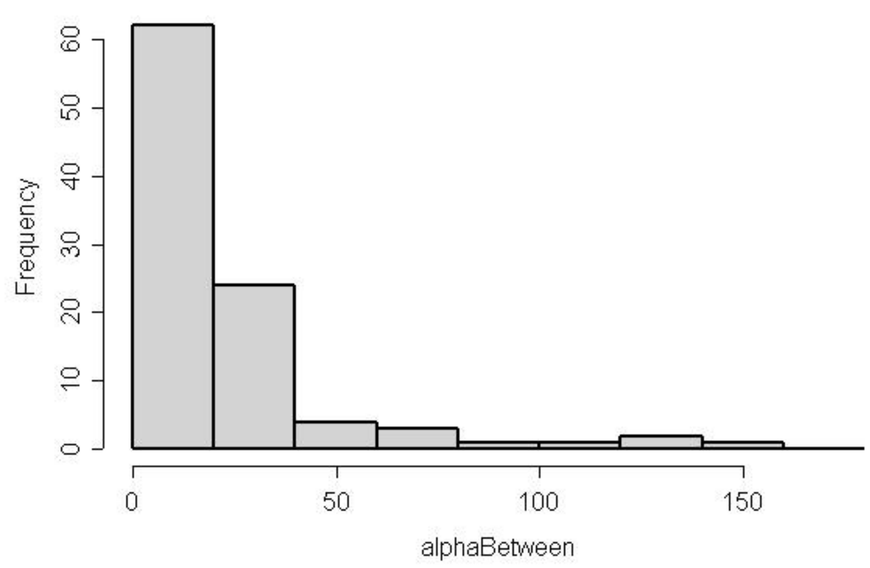

Histogram: good
Frequency
0
5
10
15
0
50
100
150
alphaMin

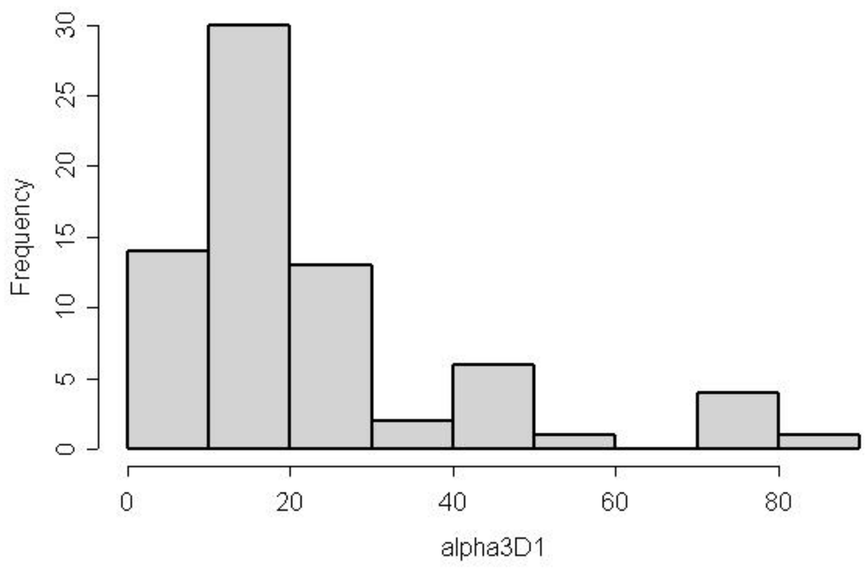

Histogram: bad
Frequency
0
5
10
15
20
25
0
50
100
150
alphaMin

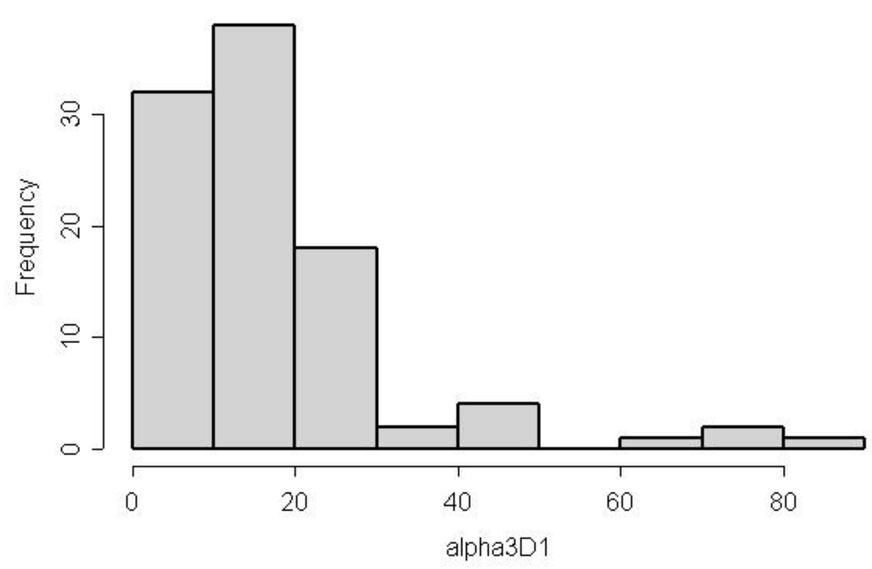

Histogram: good
Frequency
0
5
10
15
20
25
30
-200
-150
-100
-50
0
50
100
alphaMax

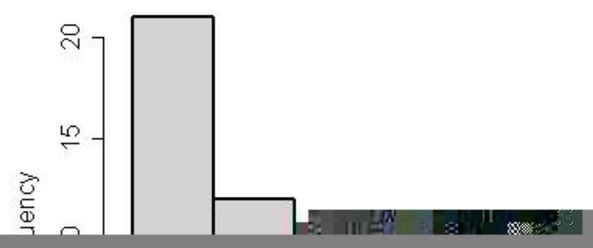

Histogram: bad
Frequency
0
10
20
30
40
50
-200
-150
-100
-50
0
50
100
alphaMax

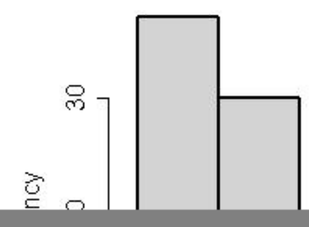

Histogram: good
Frequency
0
10
20
30
40
0
50
100
150
alphaBetween

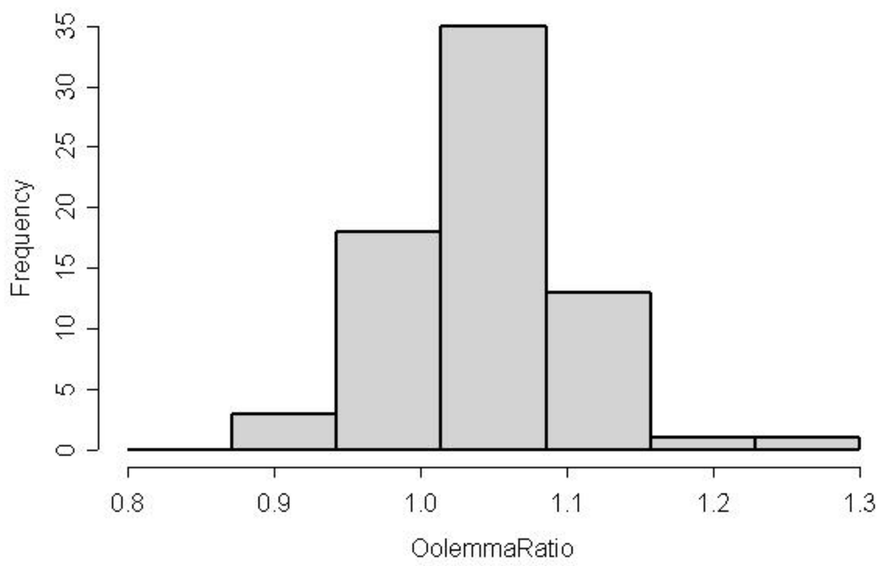

Histogram: bad
Frequency
0
10
20
30
40
50
60
0
50
100
150
alphaBetween

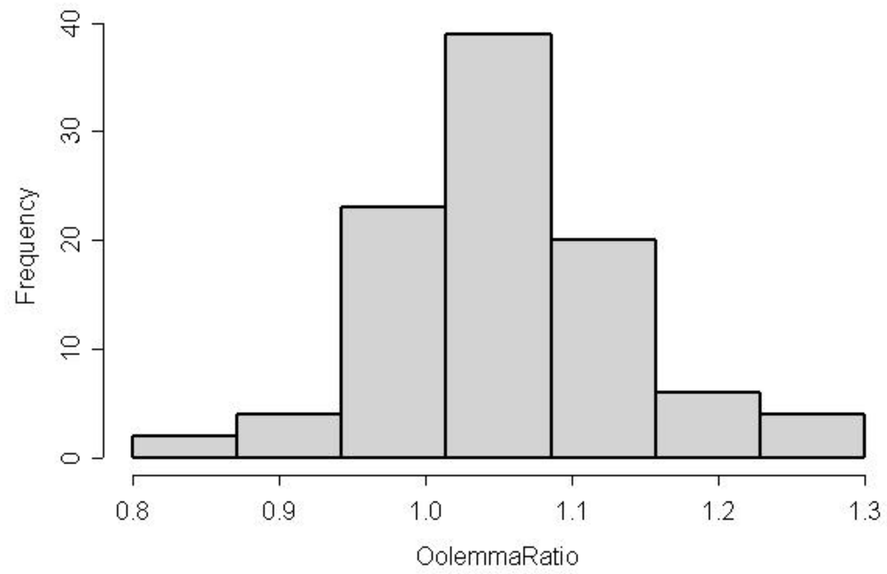

Histogram: good
Frequency
0
5
10
15
20
25
30
0
20
40
60
80
alpha3D1

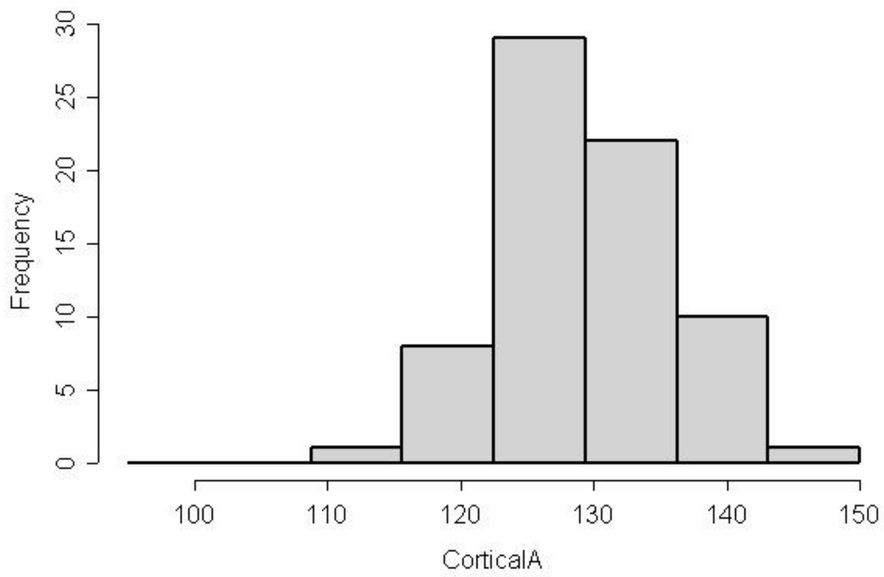

Histogram: bad
Frequency
0
10
20
30
0
20
40
60
80
alpha3D1

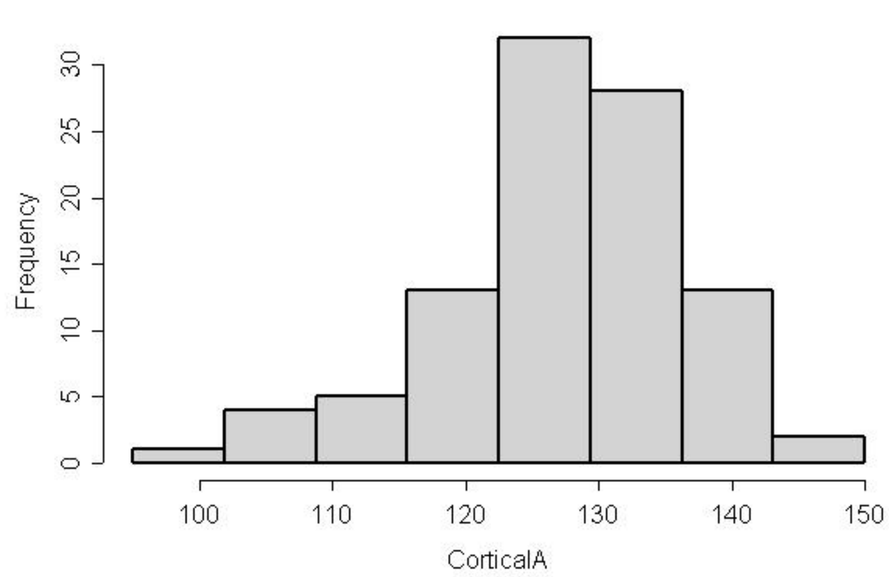

Histogram: good
20
15

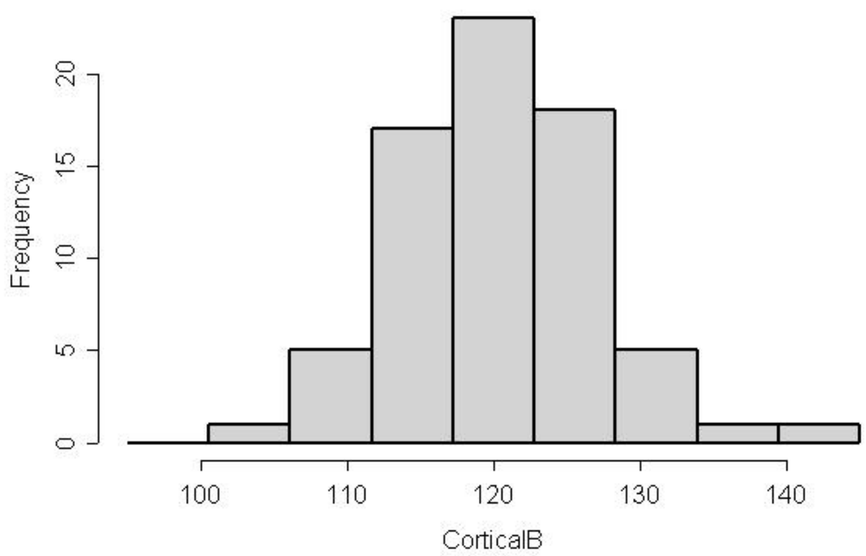

Histogram: bad
30

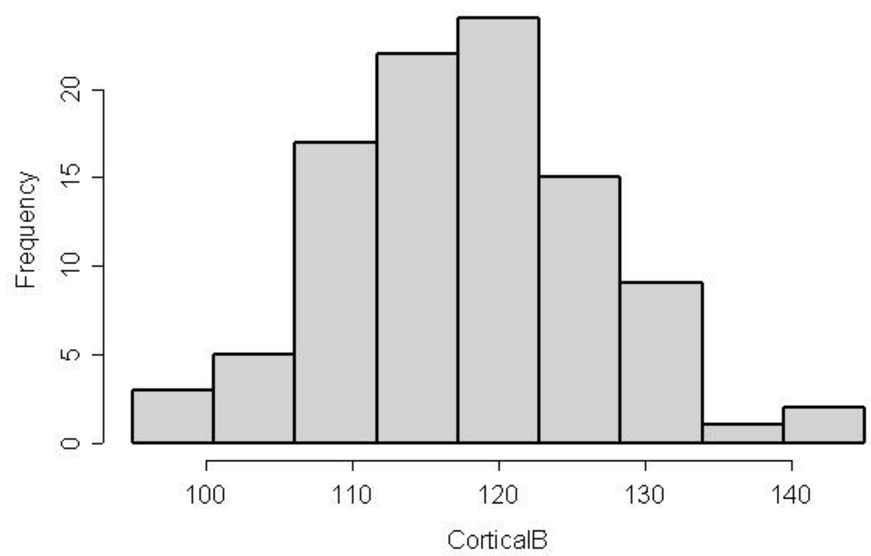

Histogram: good
Frequency
0
5
10
15
20
25
30
35
0.8
0.9
1.0
1.1
1.2
1.3
OolemmaRatio

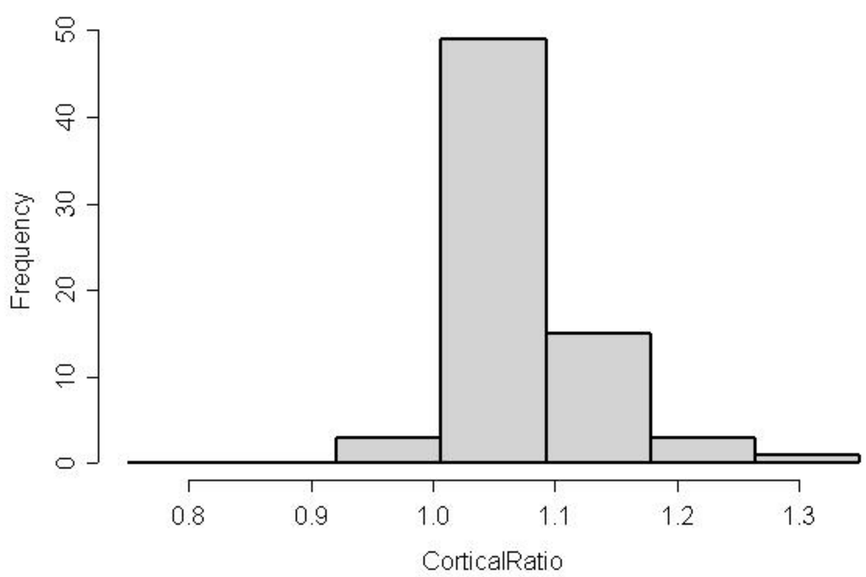

Histogram: bad
Frequency
0
10
20
30
40
0.8
0.9
1.0
1.1
1.2
1.3
OolemmaRatio

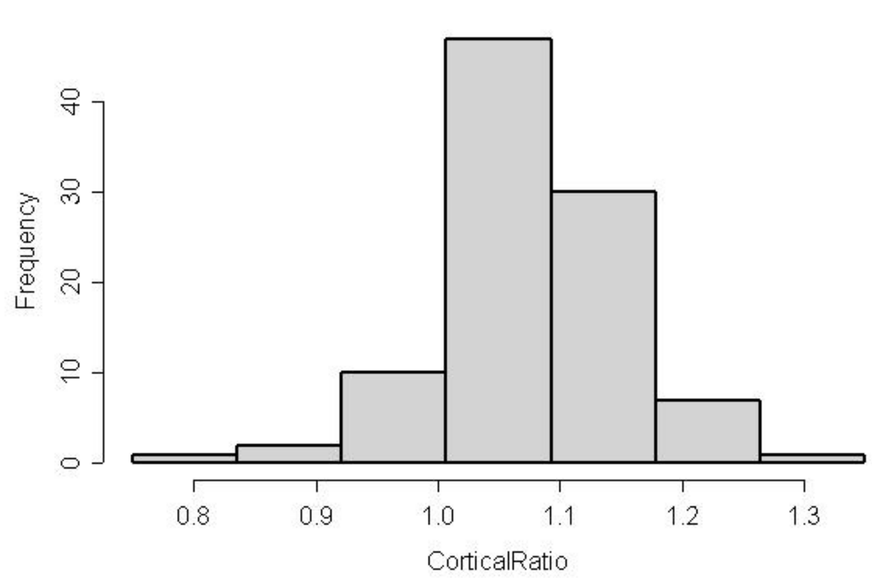

Histogram: good
Frequency
0
5
10
15
20
25
30
100
110
120
130
140
150
CorticalA

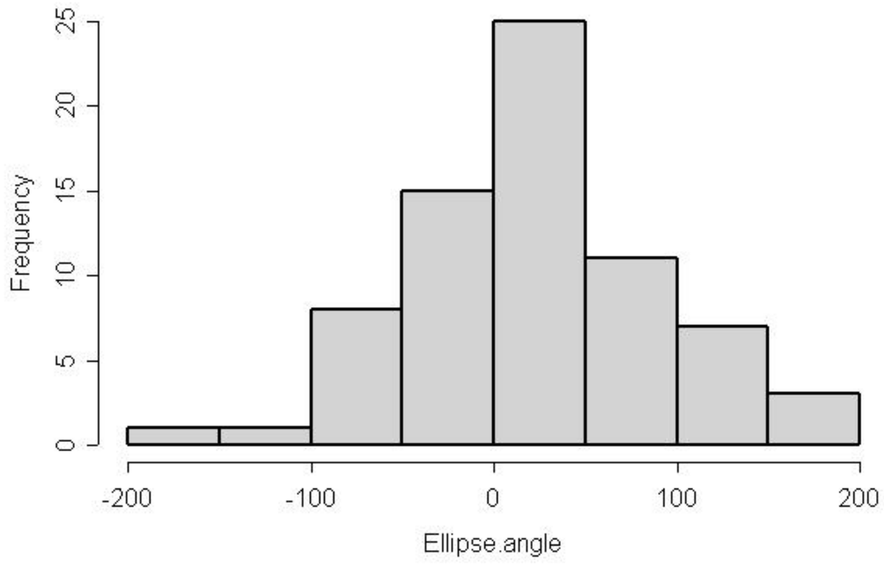

Histogram: bad
Frequency
0
5
10
15
20
25
30
100
110
120
130
140
150
CorticalA

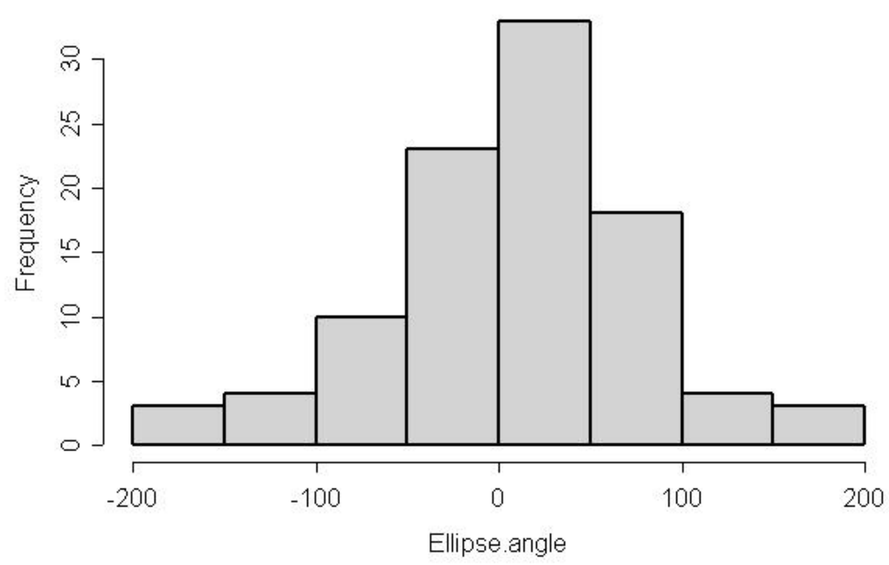

Histogram: good
Frequency
0
5
10
15
20
100
110
120
130
140
CorticalB

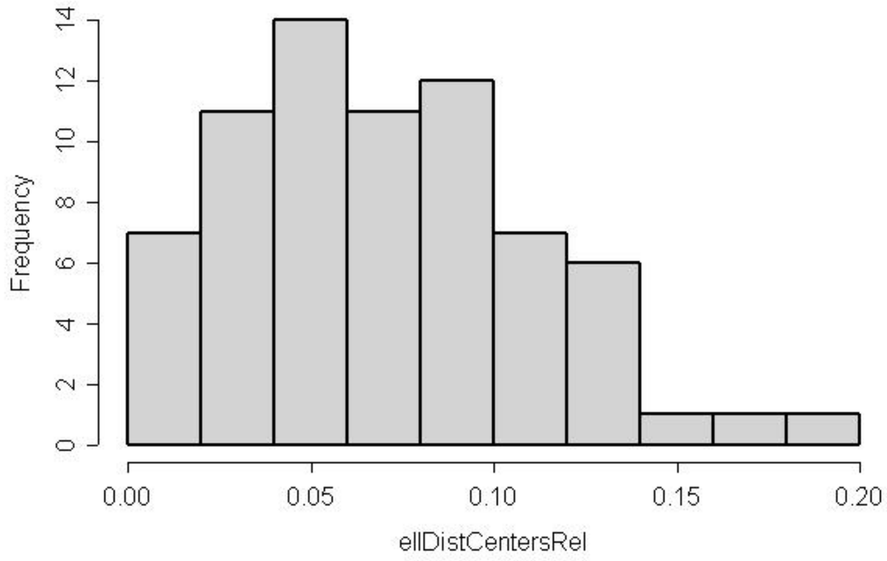

Histogram: bad
Frequency
0
5
10
15
20
100
110
120
130
140
CorticalB

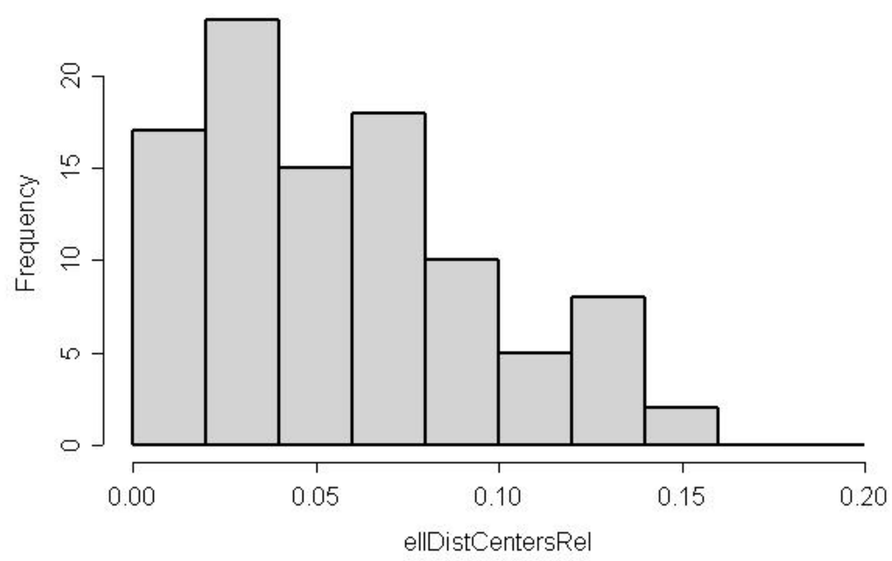

Histogram: good
Frequency
0
10
20
30
40
50
0.8
0.9
1.0
1.1
1.2
1.3
CorticalRatio

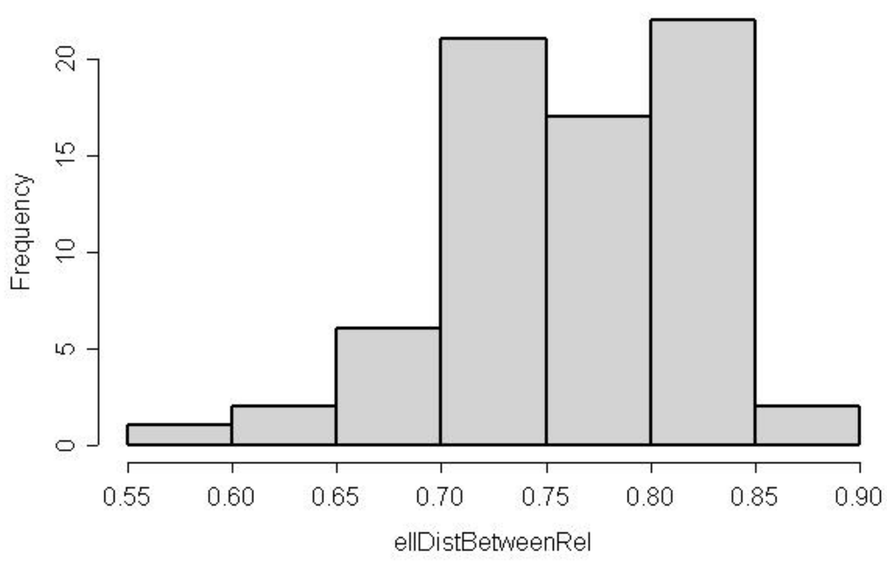

Histogram: bad
Frequency
0
10
20
30
40
0.8
0.9
1.0
1.1
1.2
1.3
CorticalRatio

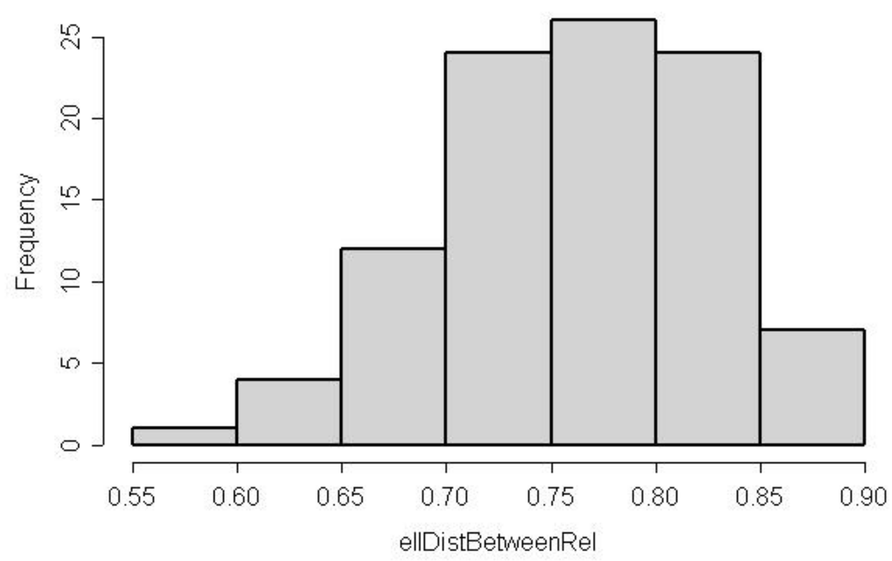

Histogram: good
Frequency
0
5
10
15
20
25
-200
-100
0
100
200
Ellipse.angle

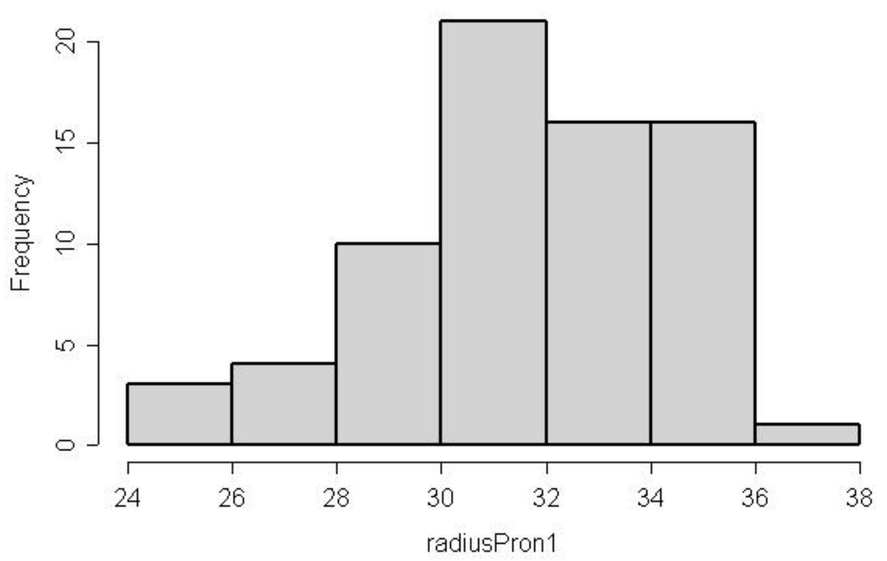

Histogram: bad
Frequency
0
5
10
15
20
25
30
-200
-100
0
100
200
Ellipse.angle

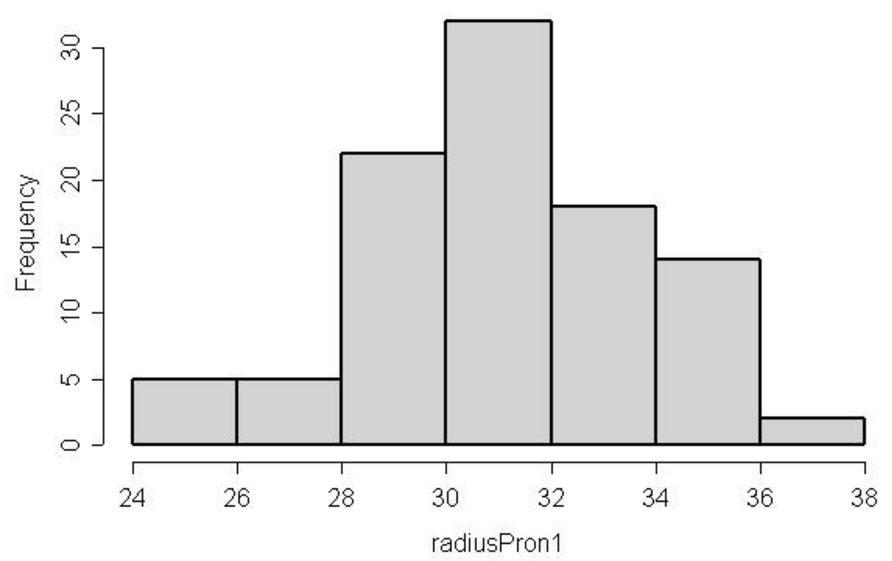

Histogram: good
Frequency
0
2
4
6
8
10
12
14
0.00
0.05
0.10
0.15
0.20
ellDistCentersRel

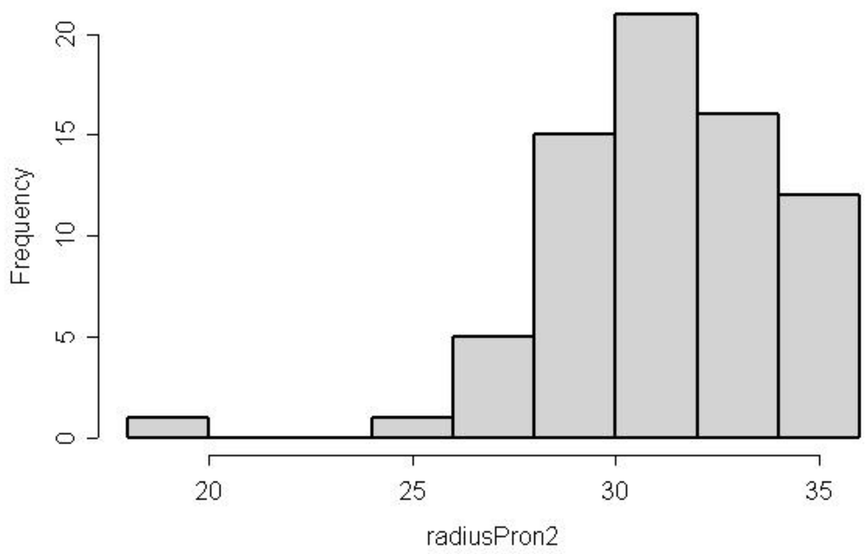

Histogram: bad
Frequency
0
5
10
15
20
0.00
0.05
0.10
0.15
0.20
ellDistCentersRel

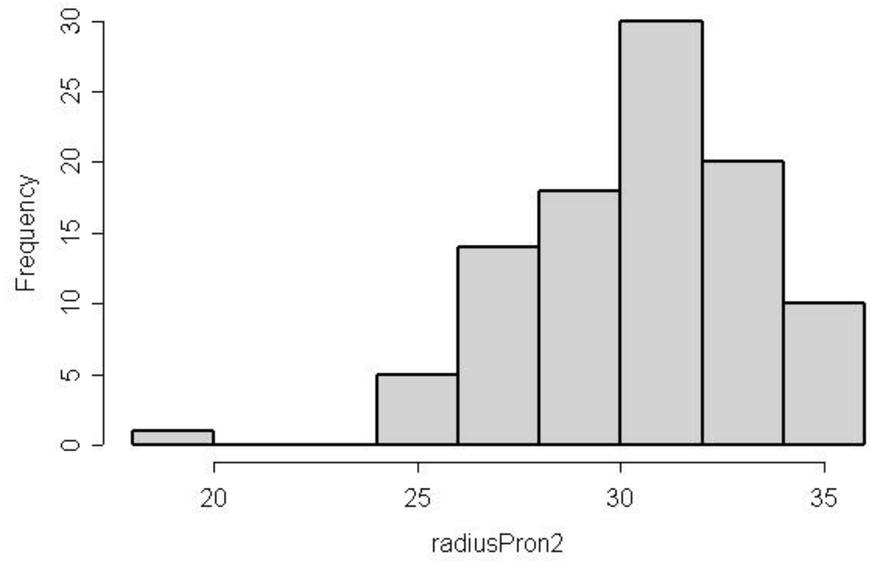

Histogram: good
Frequency
0
5
10
15
20
0.55
0.60
0.65
0.70
0.75
0.80
0.85
0.90
ellDistBetweenRel

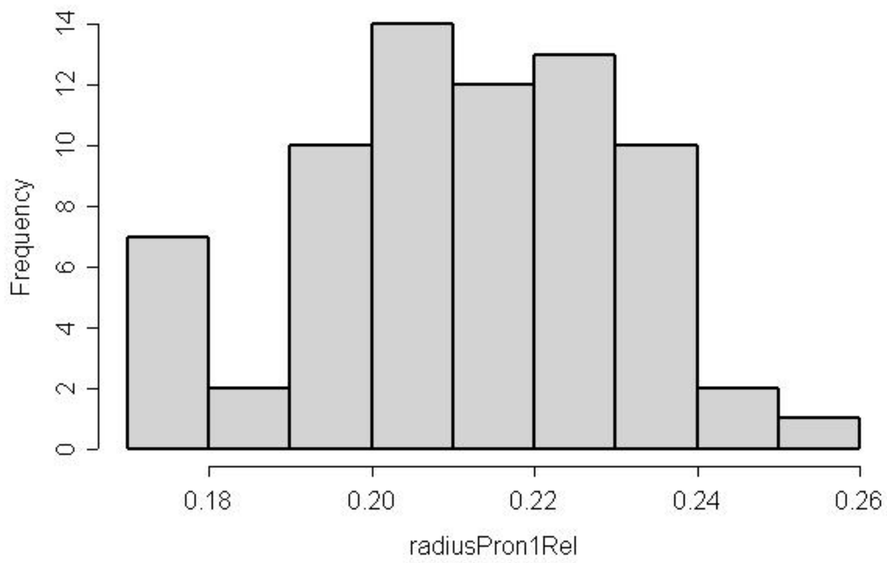

Histogram: bad
Frequency
0
5
10
15
20
25
0.55
0.60
0.65
0.70
0.75
0.80
0.85
0.90
ellDistBetweenRel

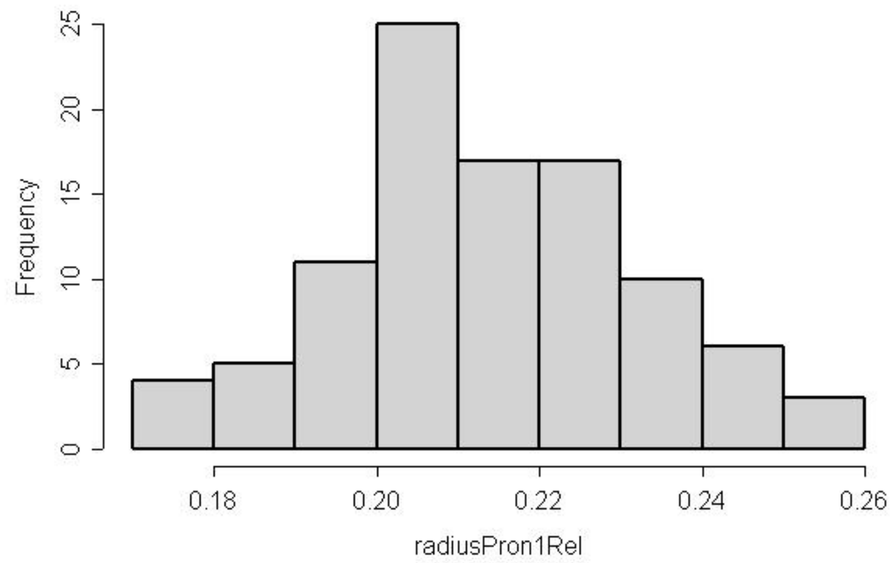

Histogram: good
Frequency
0
5
10
15
20
24
26
28
30
32
34
36
38
radiusPron1

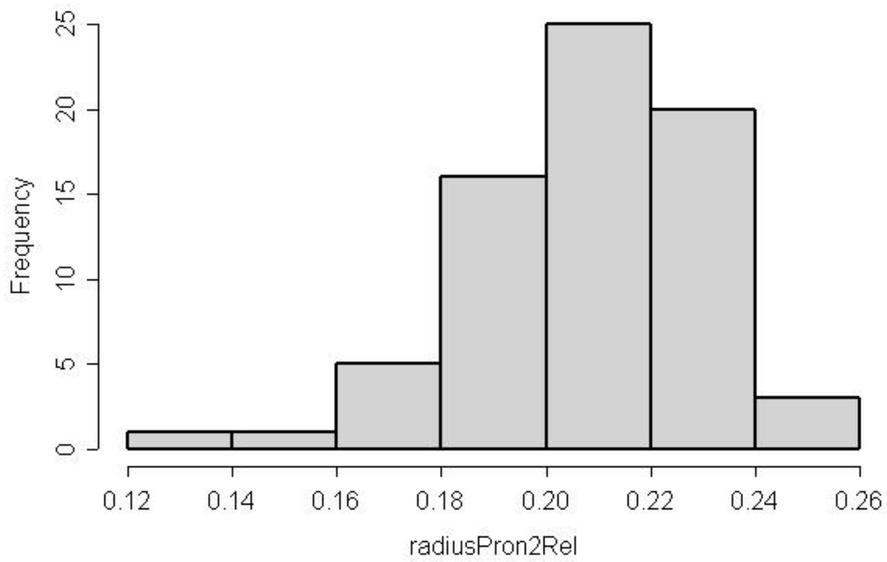

Histogram: bad
Frequency
0
5
10
15
20
25
30
24
26
28
30
32
34
36
38
radiusPron1

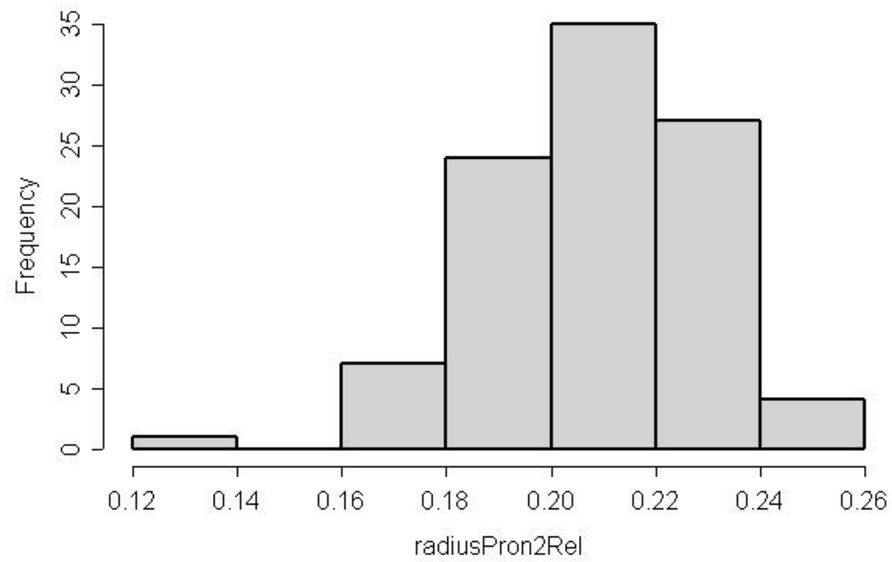

Histogram: good
Frequency
0
5
10
15
20
20
25
30
35
radiusPron2

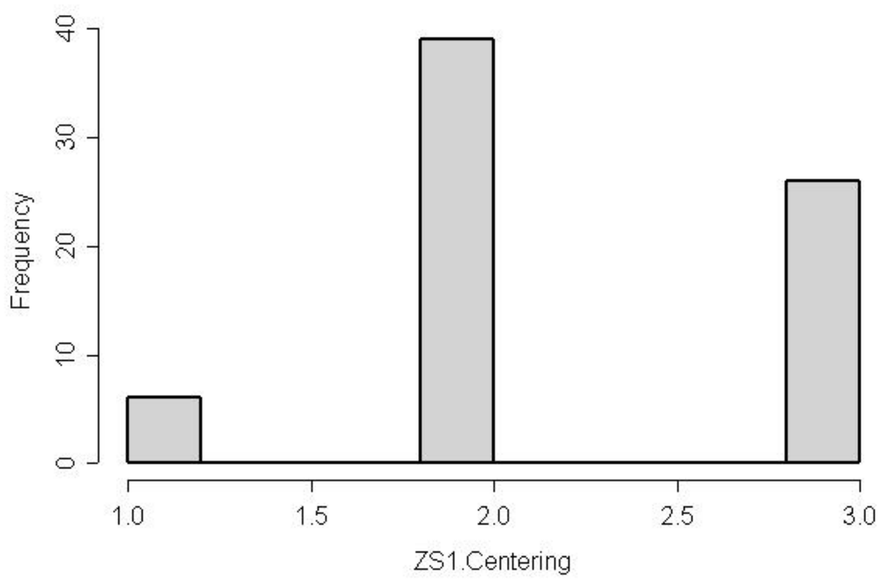

Histogram: bad
Frequency
0
5
10
15
20
25
30
20
25
30
35
radiusPron2

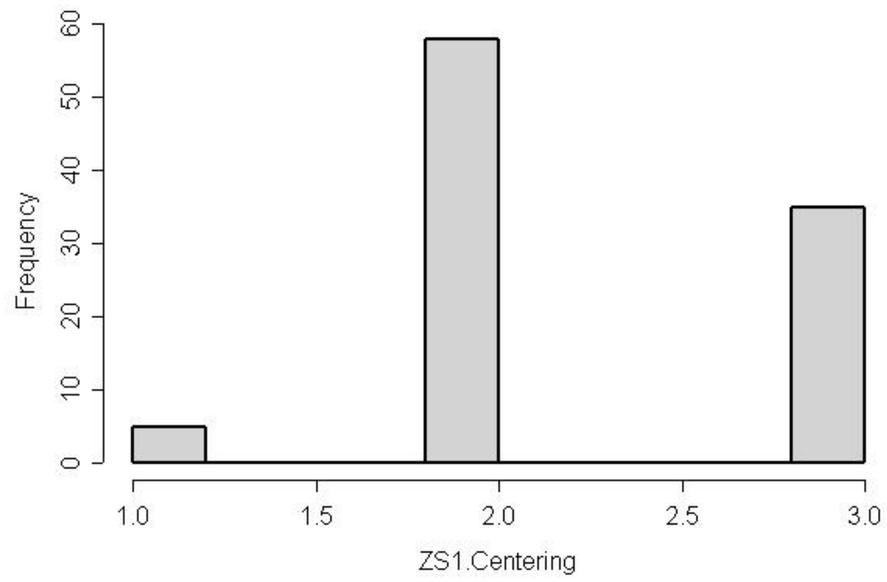

Histogram: good
Frequency
0
2
4
6
8
10
12
14
0.18
0.20
0.22
0.24
0.26
radiusPron1Rel

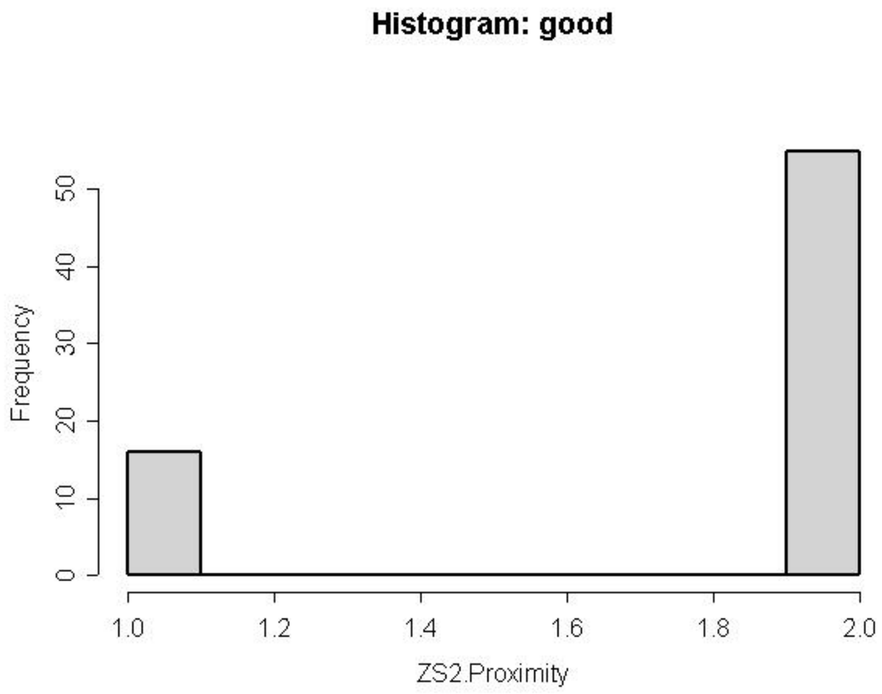

Histogram: bad
Frequency
0
5
10
15
20
25
0.18
0.20
0.22
0.24
0.26
radiusPron1Rel

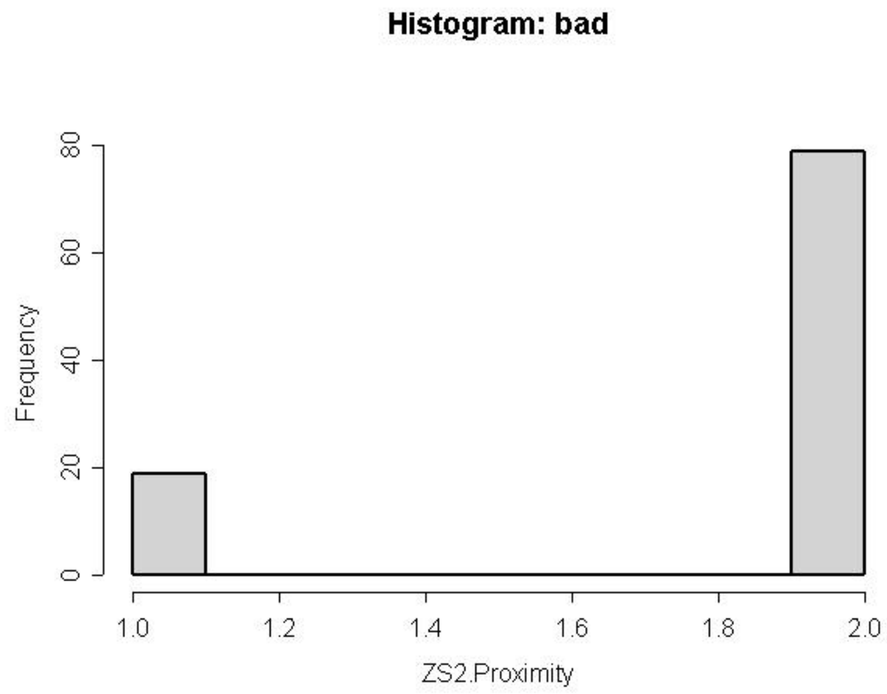

Histogram: good
Frequency
0
5
10
15
20
25
0.12
0.14
0.16
0.18
0.20
0.22
0.24
0.26
radiusPron2Rel

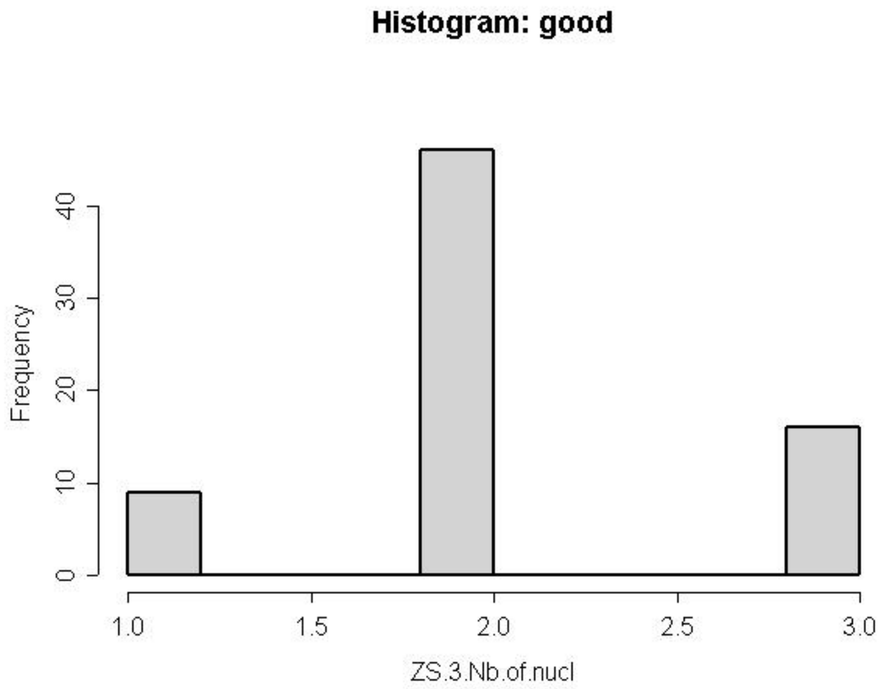

Histogram: bad
Frequency
0
5
10
15
20
25
30
35
0.12
0.14
0.16
0.18
0.20
0.22
0.24
0.26
radiusPron2Rel

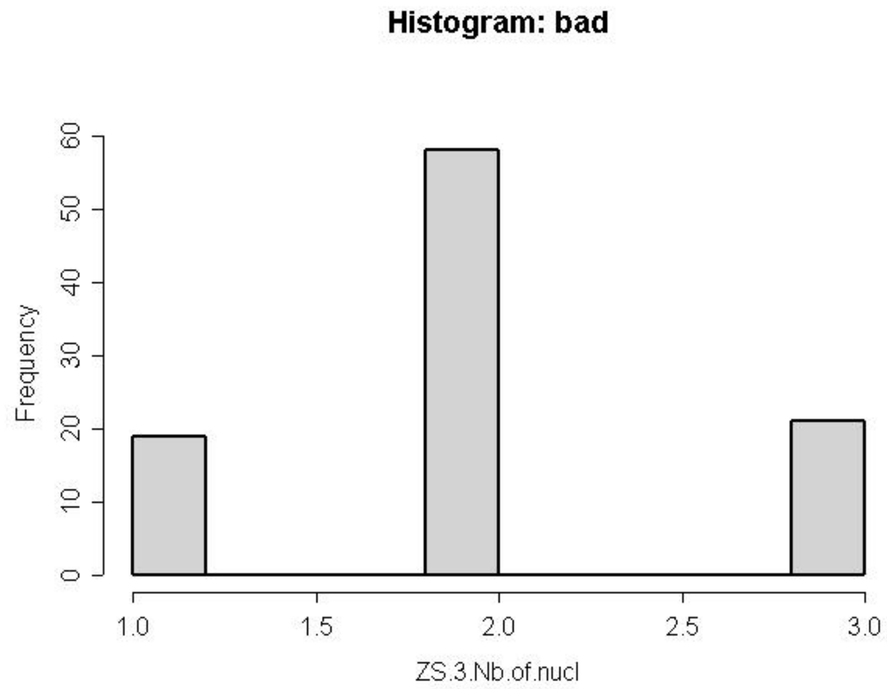

Histogram: good
Frequency
0
10
20
30
40
1.0
1.5
2.0
2.5
3.0
ZS1.Centering

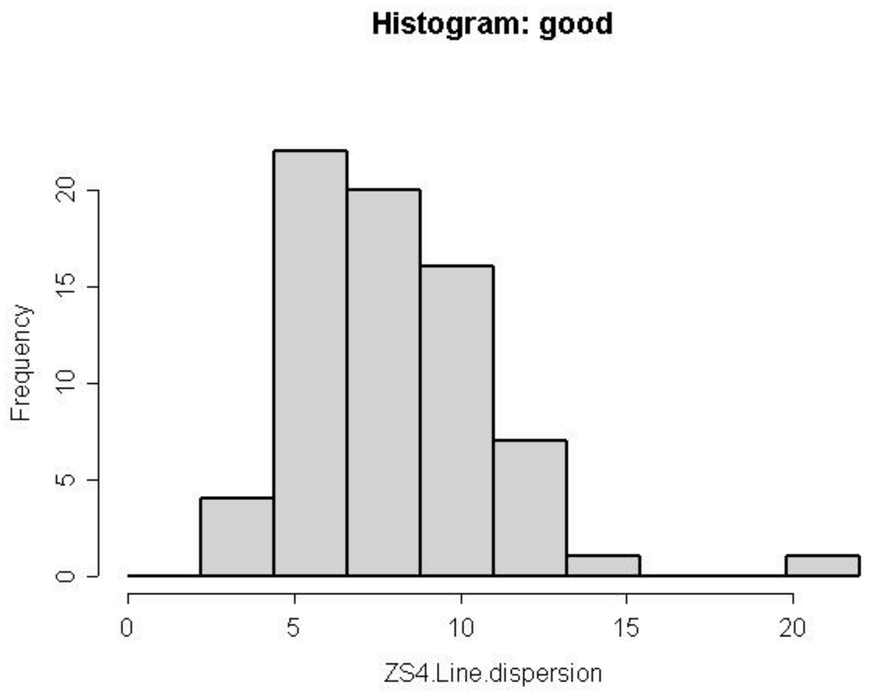

Histogram: bad
Frequency
0
10
20
30
40
50
60
1.0
1.5
2.0
2.5
3.0
ZS1.Centering

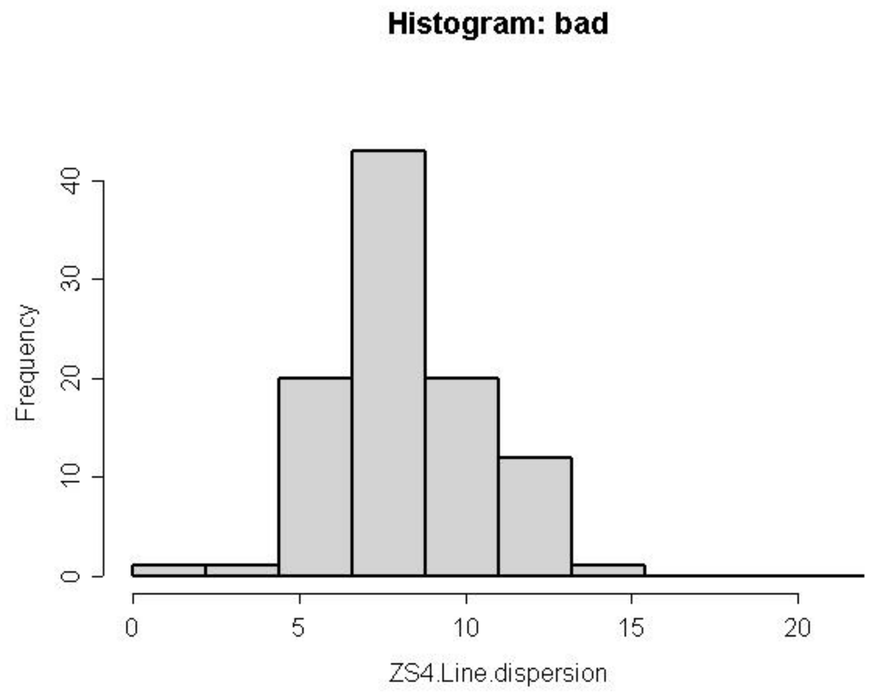

Histogram: good
Frequency
0
10
20
30
40
50
1.0
1.2
1.4
1.6
1.8
2.0
ZS2.Proximity

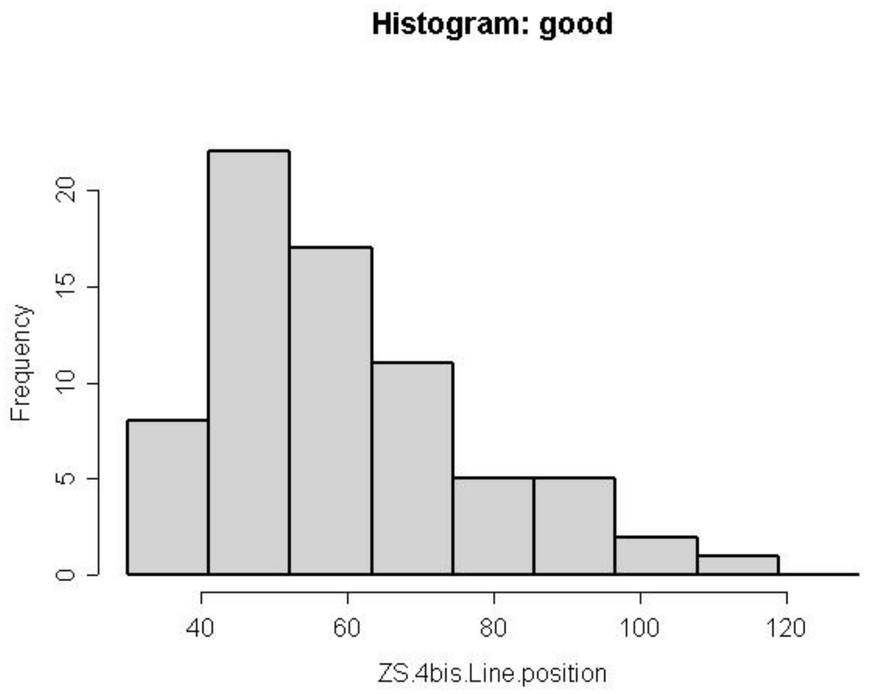

Histogram: bad
Frequency
0
20
40
60
80
1.0
1.2
1.4
1.6
1.8
2.0
ZS2.Proximity

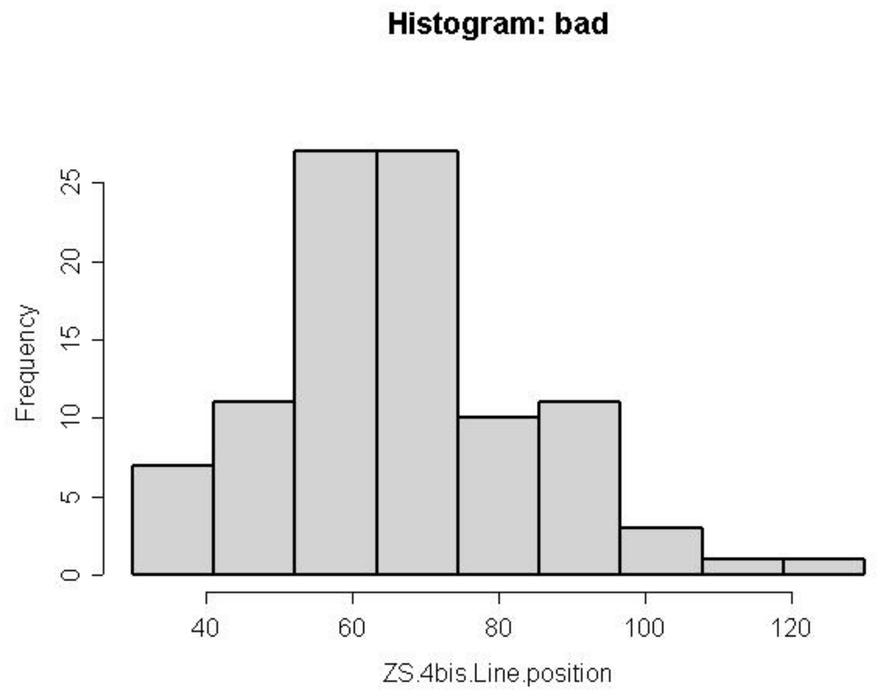

Histogram: good
Frequency
0
10
20
30
40
1.0
1.5
2.0
2.5
3.0
ZS.3.Nb.of.nucl

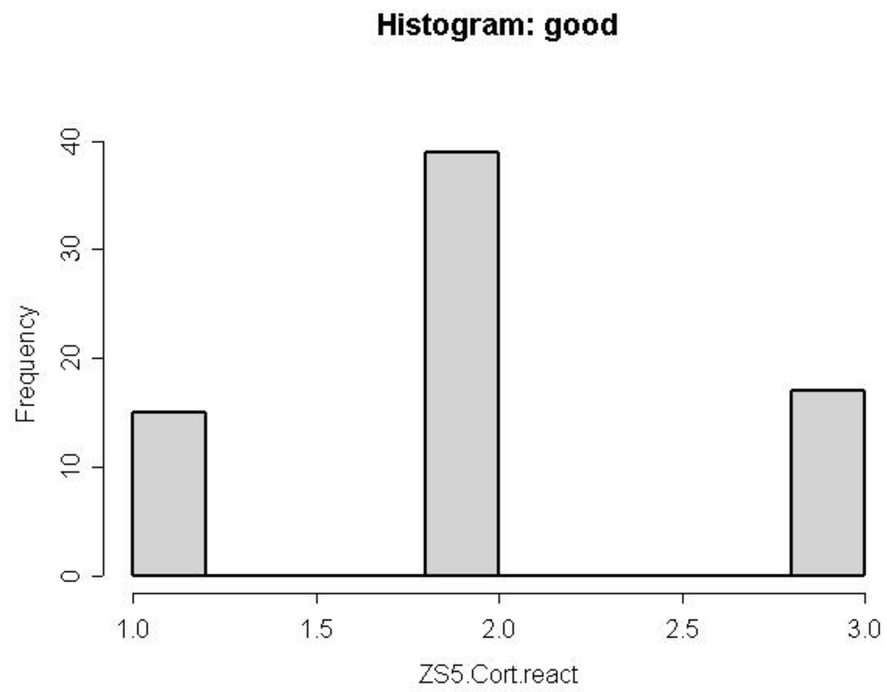

Histogram: bad
Frequency
0
10
20
30
40
50
60
1.0
1.5
2.0
2.5
3.0
ZS.3.Nb.of.nucl

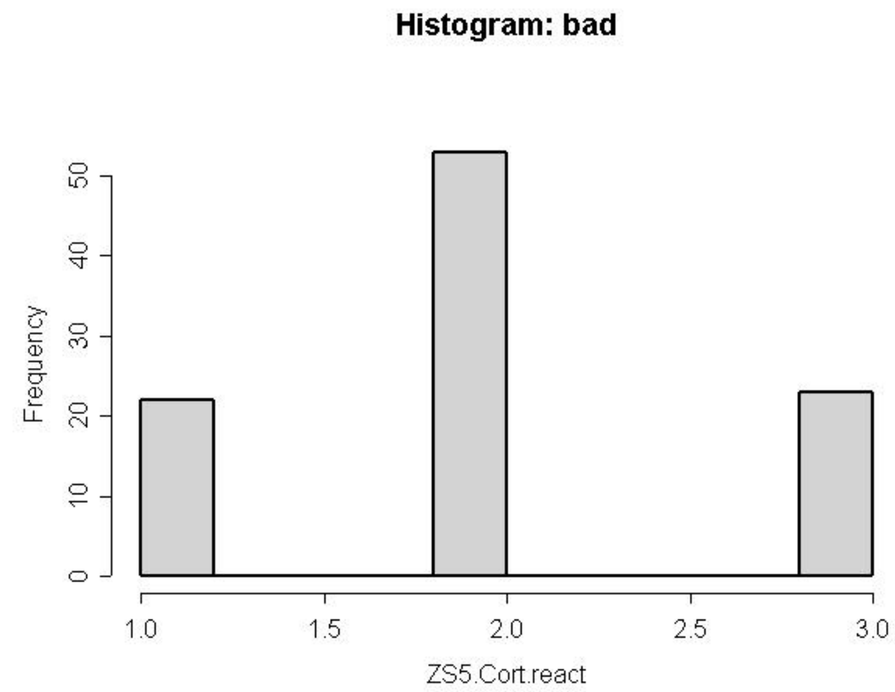

Histogram: good
Frequency
0
5
10
15
20
0
5
10
15
20
ZS4.Line.dispersion

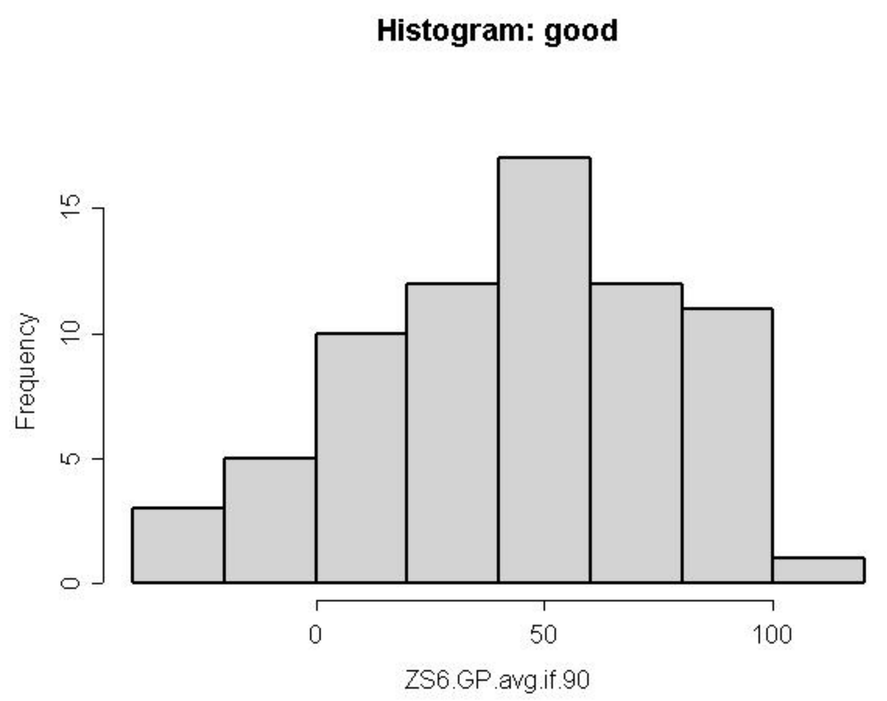

Histogram: bad
Frequency
0
10
20
30
40
0
5
10
15
20
ZS4.Line.dispersion

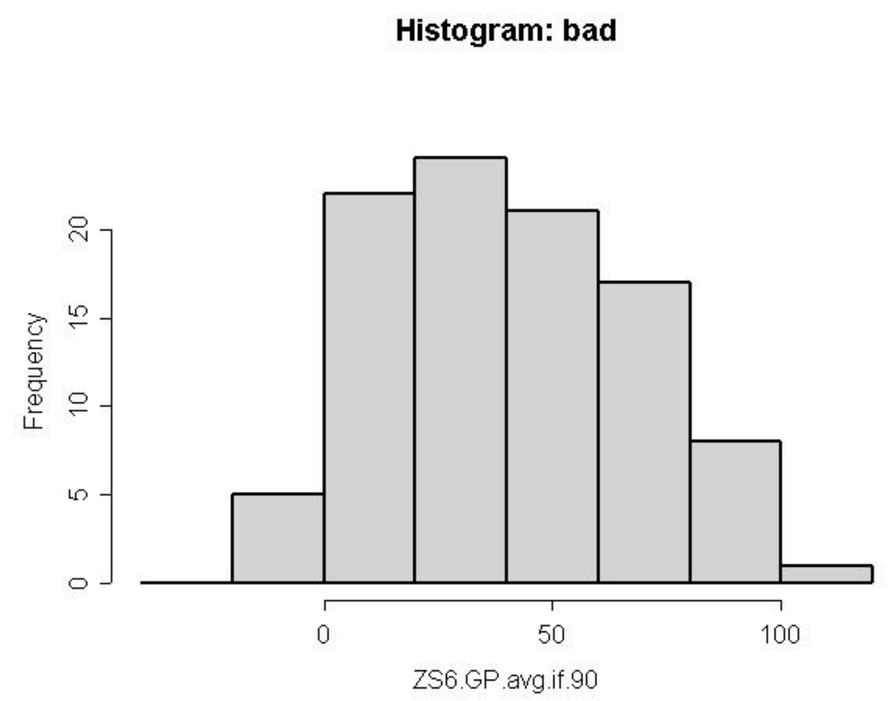

Histogram: good
Frequency
0
5
10
15
20
40
60
80
100
120
ZS.4bis.Line.position

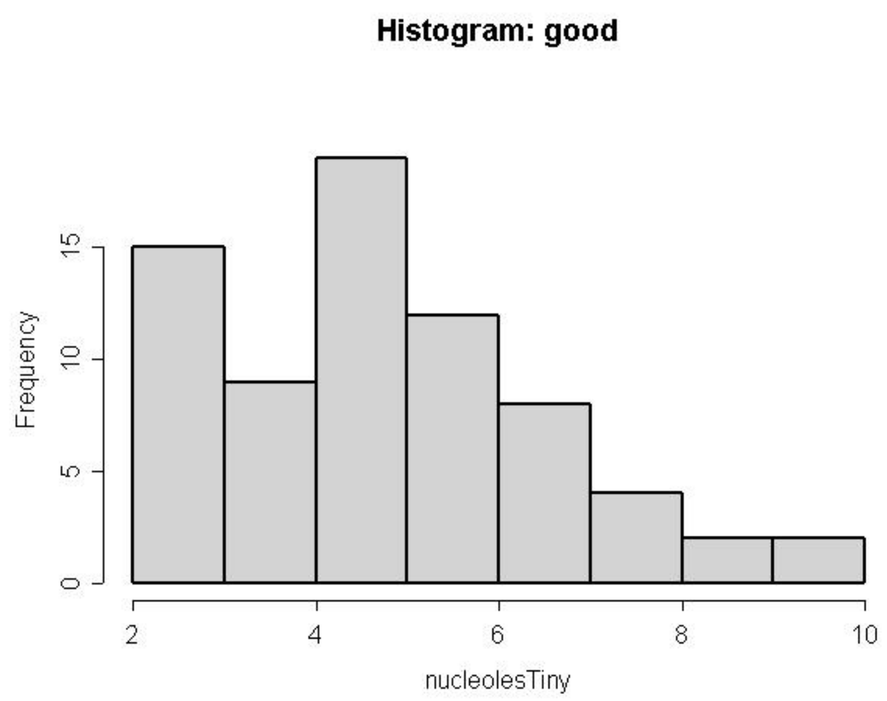

Histogram: bad
Frequency
0
5
10
15
20
25
40
60
80
100
120
ZS.4bis.Line.position

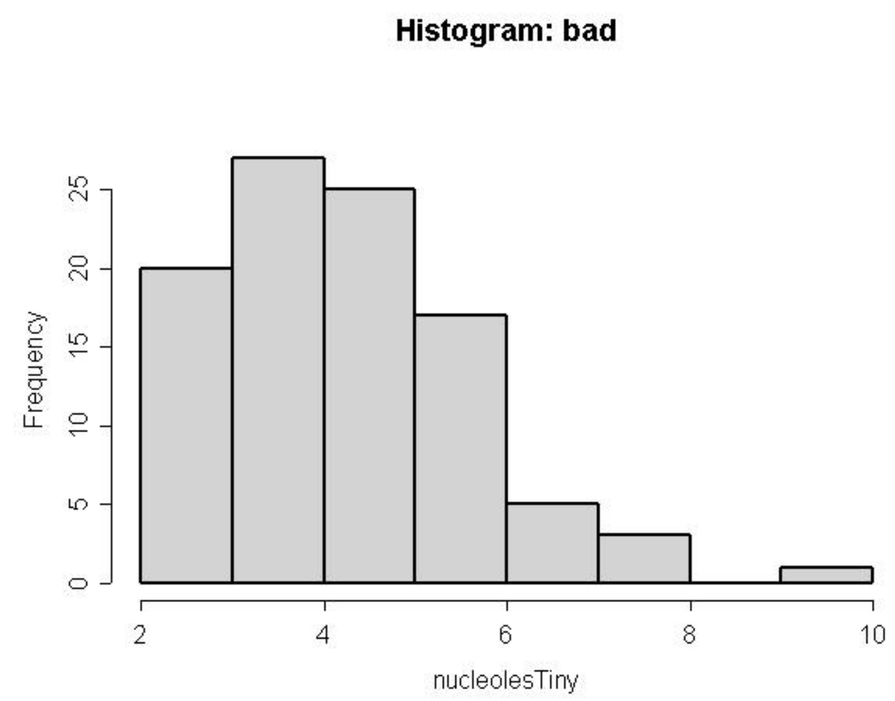

Histogram: good
Frequency
0
10
20
30
40
1.0
1.5
2.0
2.5
3.0
ZS5.Cort.react

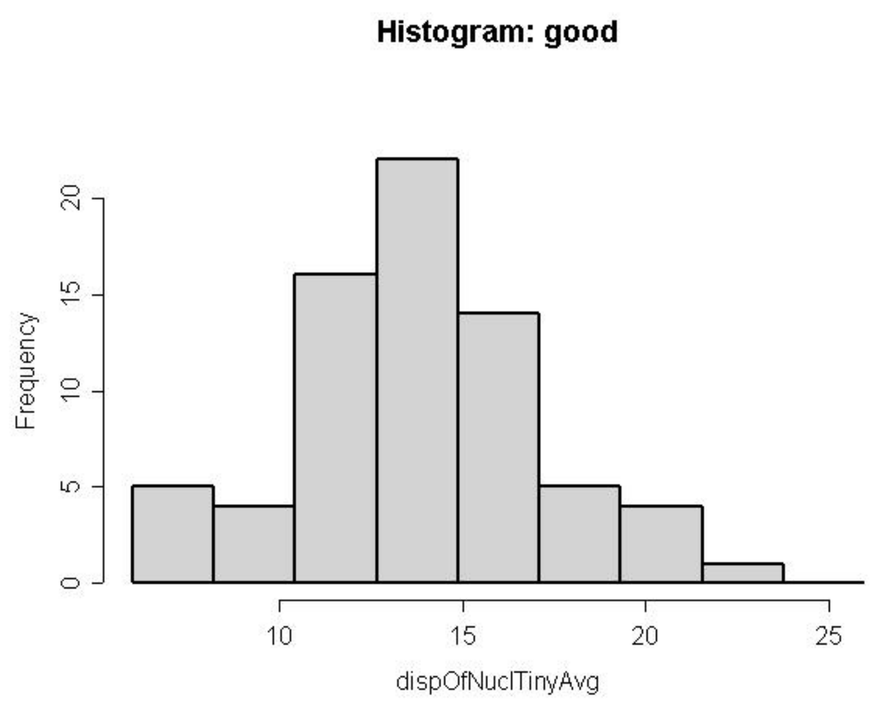

Histogram: bad
Frequency
0
10
20
30
40
50
1.0
1.5
2.0
2.5
3.0
ZS5.Cort.react

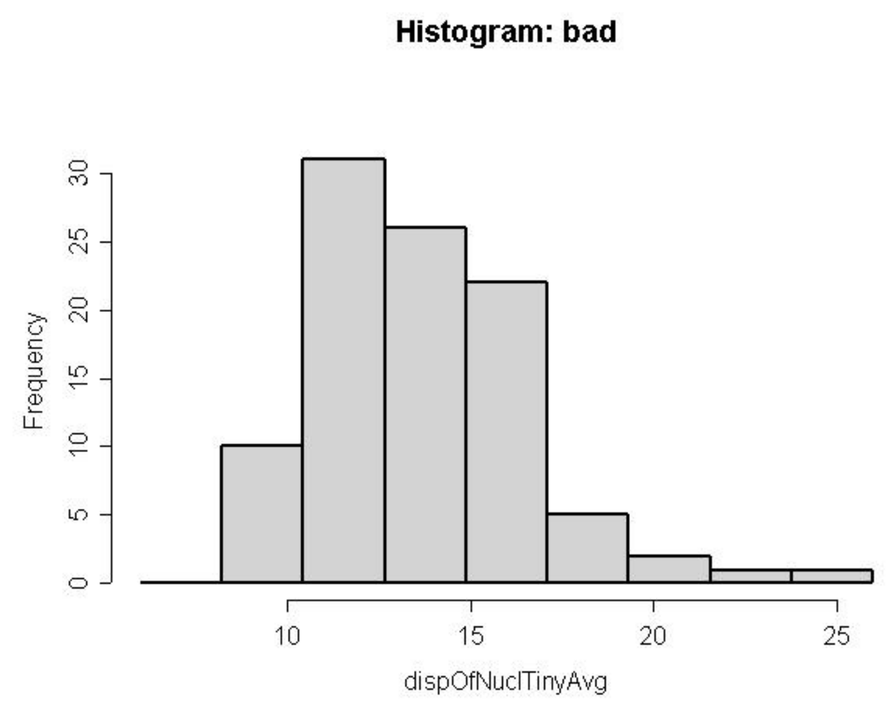

Histogram: good
Frequency
0
5
10
15
0
50
100
ZS6.GP.avg.if.90

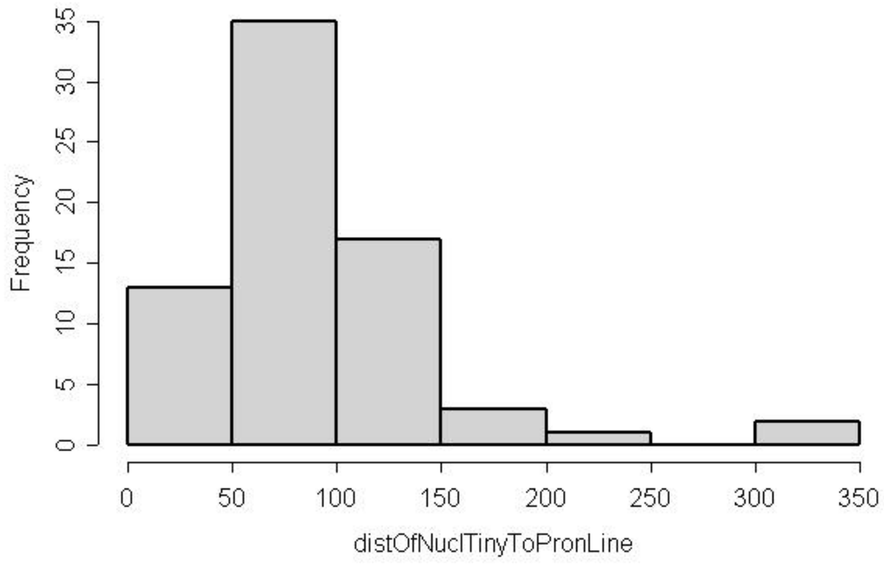

Histogram: bad
Frequency
0
5
10
15
20
0
50
100
ZS6.GP.avg.if.90

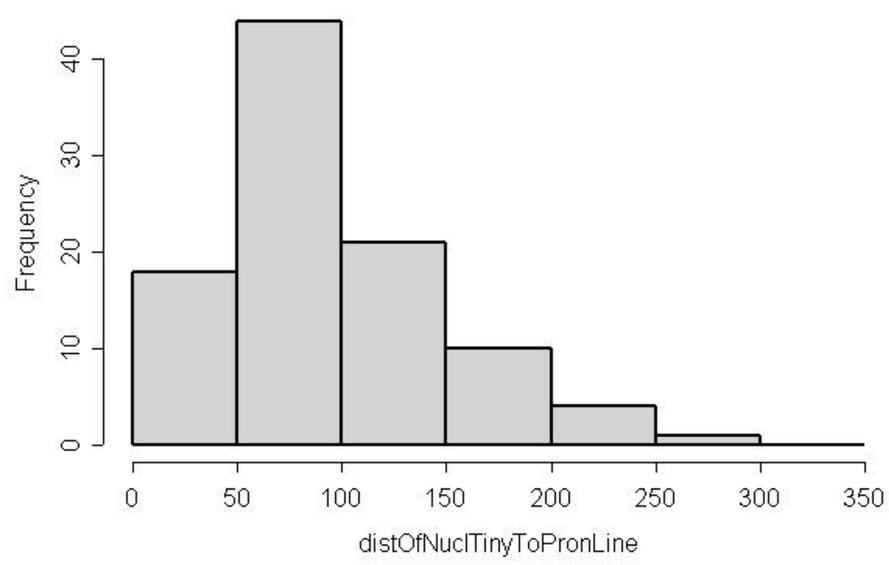

Histogram: good
Frequency
0
5
10
15
2
4
6
8
10
nucleolesTiny

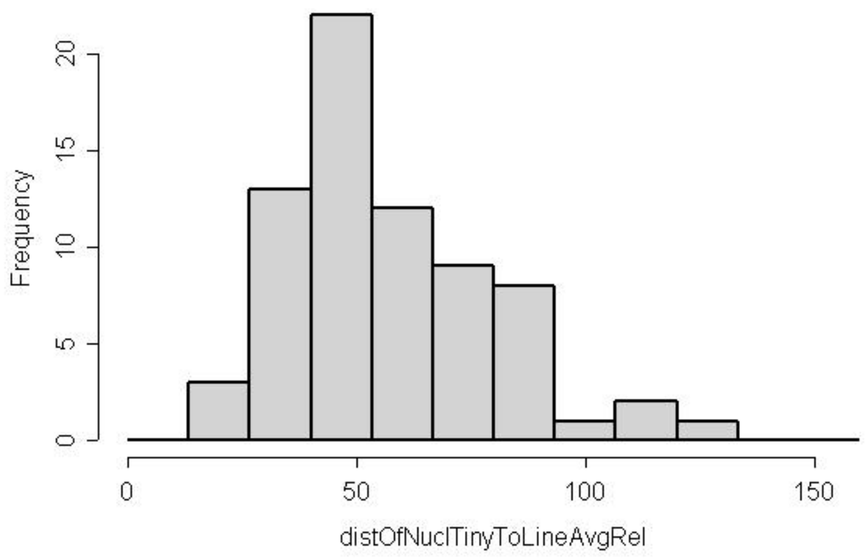

Histogram: bad
Frequency
0
5
10
15
20
25
2
4
6
8
10
nucleolesTiny

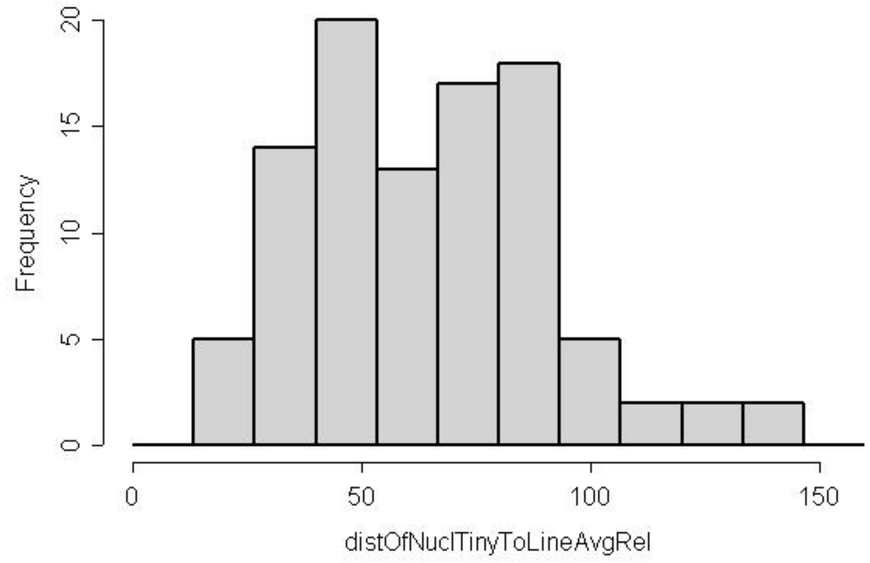

Histogram: good
Frequency
0
5
10
15
20
10
15
20
25
dispOfNuclTinyAvg

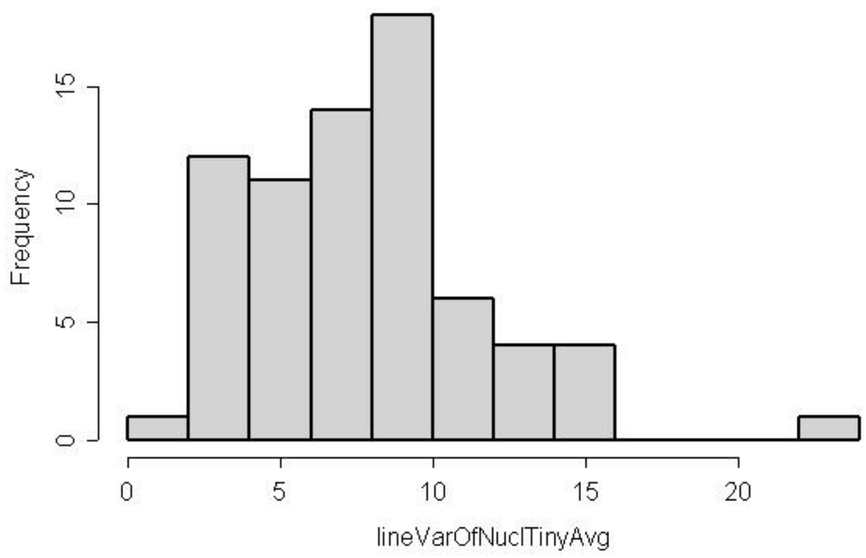

Histogram: bad
Frequency
0
5
10
15
20
25
30
10
15
20
25
dispOfNuclTinyAvg

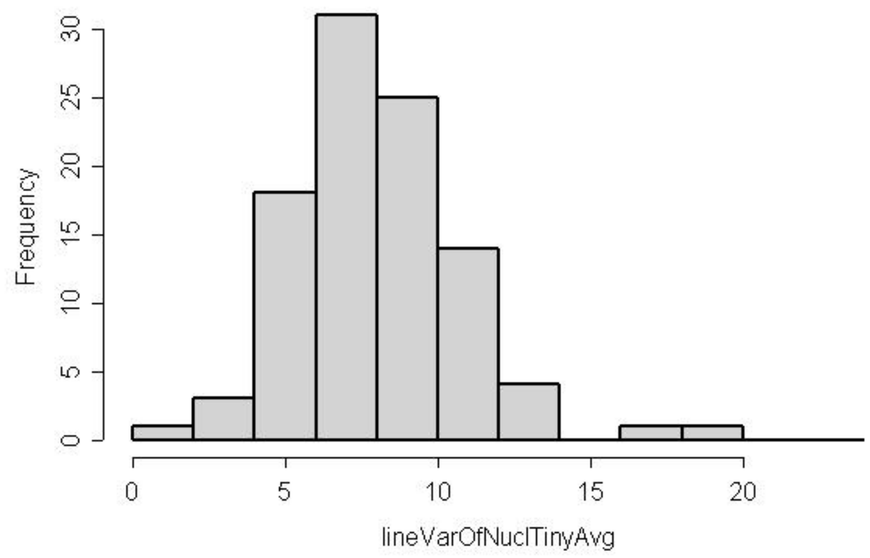

Histogram: good
Frequency
0
5
10
15
20
25
30
35
0
50
100
150
200
250
300
350
distOfNuclTinyToPronLine

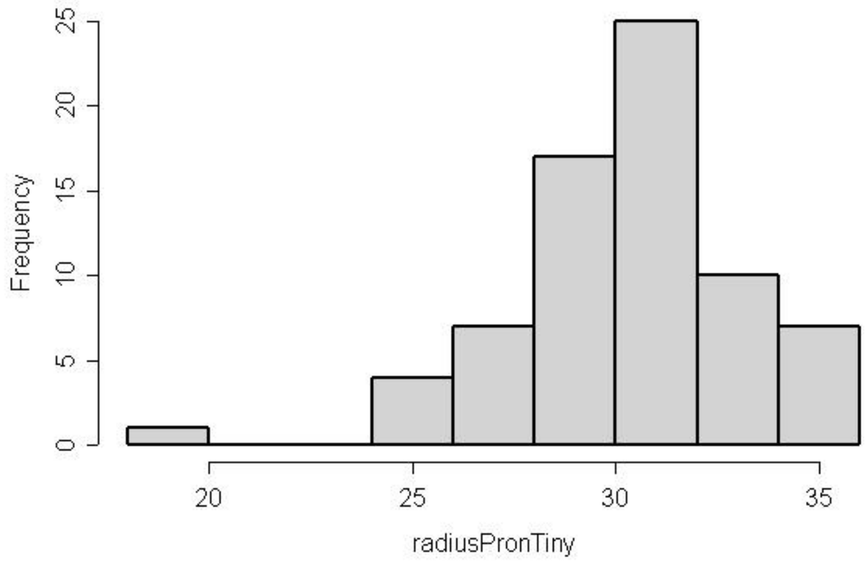

Histogram: bad
Frequency
0
10
20
30
40
0
50
100
150
200
250
300
350
distOfNuclTinyToPronLine

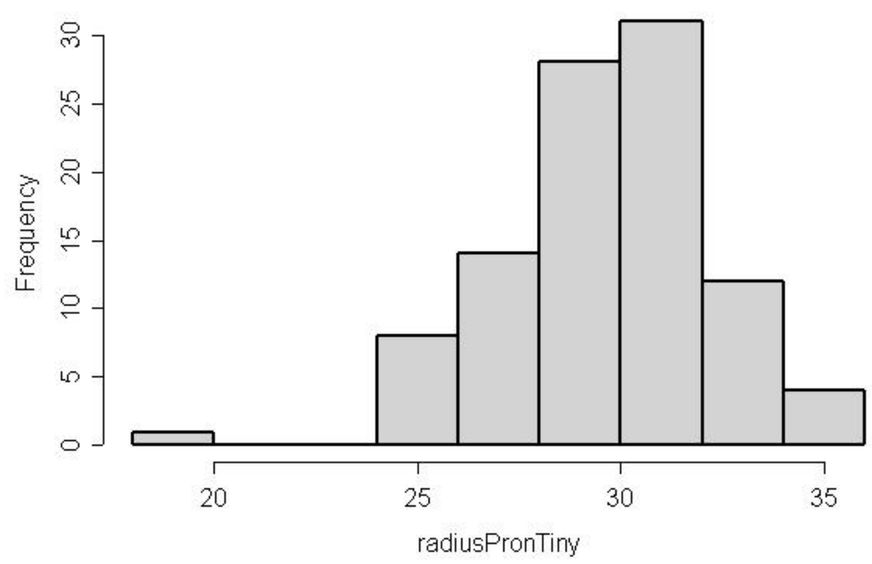

Histogram: good
Frequency
0
5
10
15
20
0
50
100
150
distOfNuclTinyToLineAvgRel

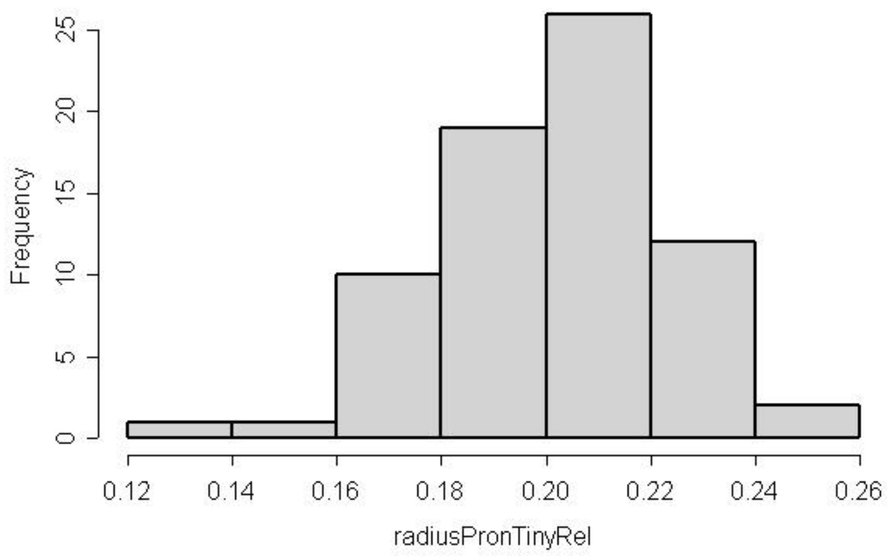

Histogram: bad
Frequency
0
5
10
15
20
0
50
100
150
distOfNuclTinyToLineAvgRel

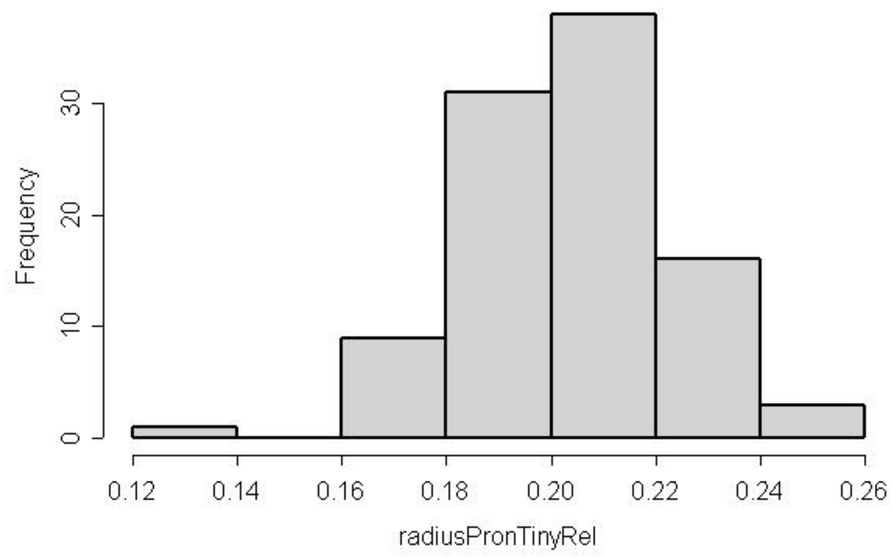

Histogram: good
Frequency
0
5
10
15
0
5
10
15
20
lineVarOfNuclTinyAvg

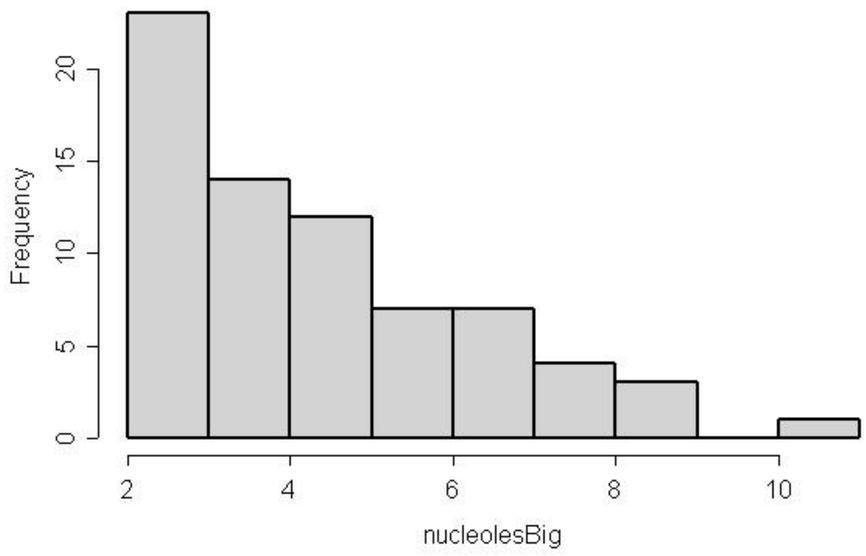

Histogram: bad
Frequency
0
5
10
15
20
25
30
0
5
10
15
20
lineVarOfNuclTinyAvg

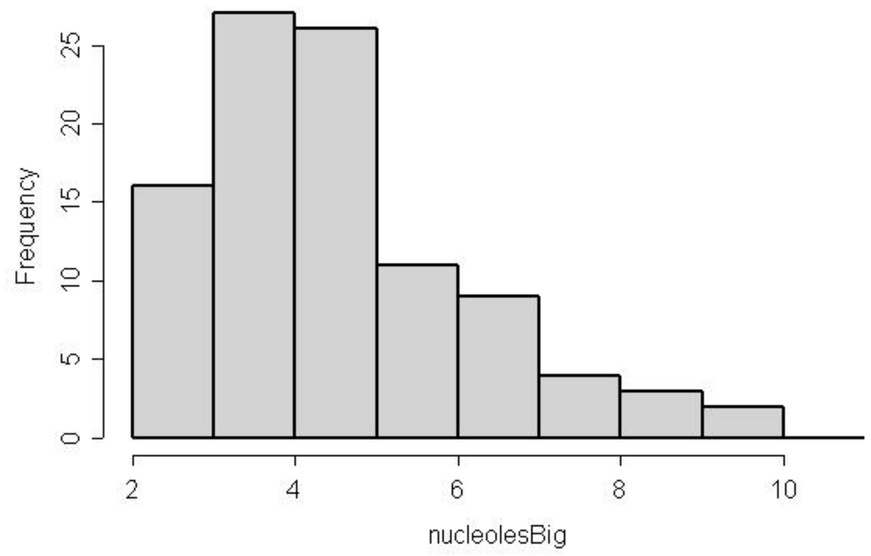

Histogram: good
Frequency
0
5
10
15
20
25
20
25
30
35
radiusPronTiny

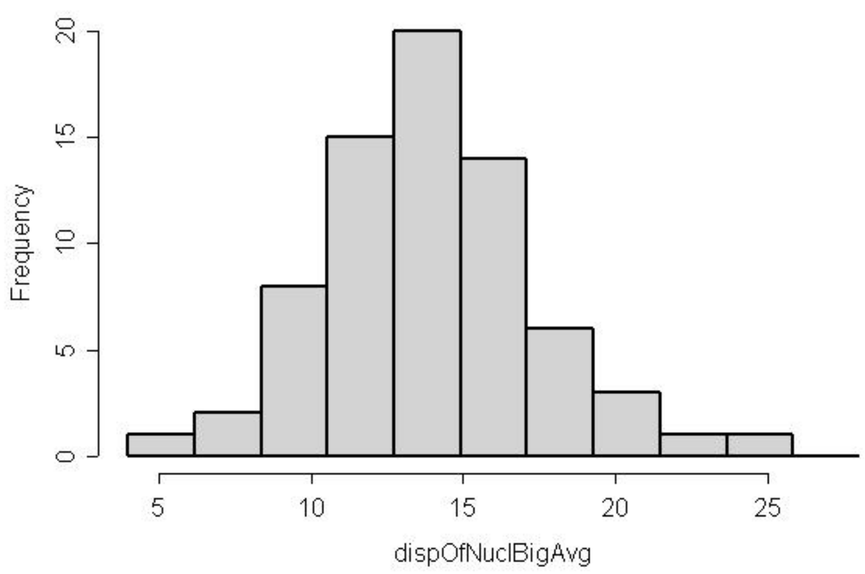

Histogram: bad
Frequency
0
5
10
15
20
25
30
20
25
30
35
radiusPronTiny

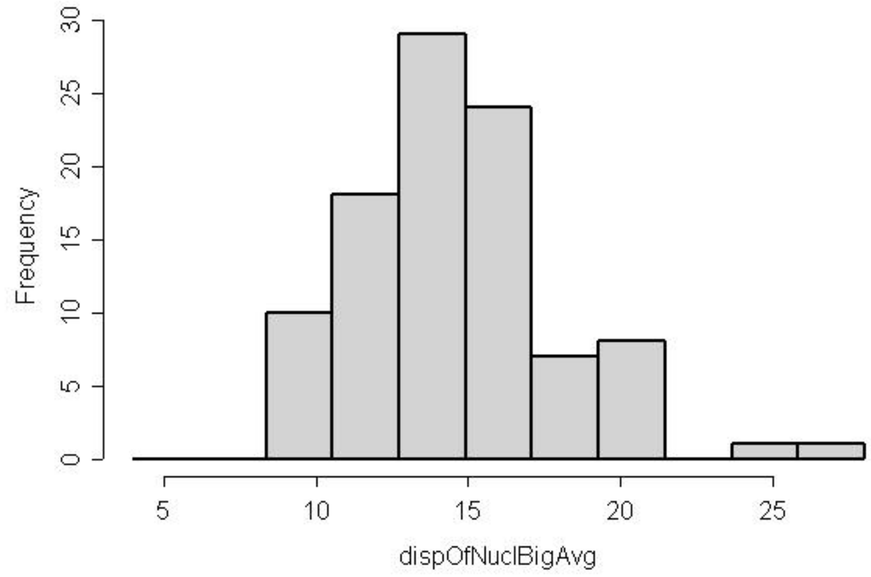

Histogram: good
Frequency
0
5
10
15
20
25
0.12
0.14
0.16
0.18
0.20
0.22
0.24
0.26
radiusPronTinyRel

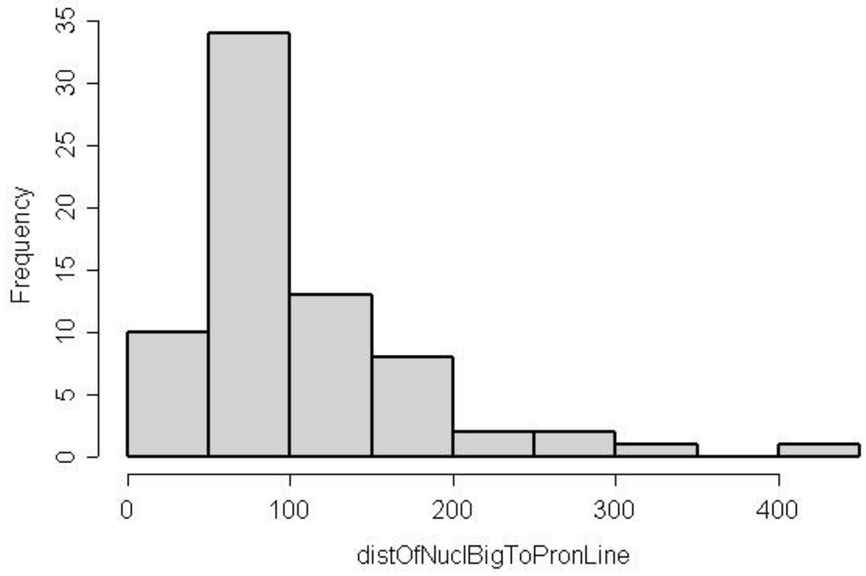

Histogram: bad
Frequency
0
10
20
30
0.12
0.14
0.16
0.18
0.20
0.22
0.24
0.26
radiusPronTinyRel

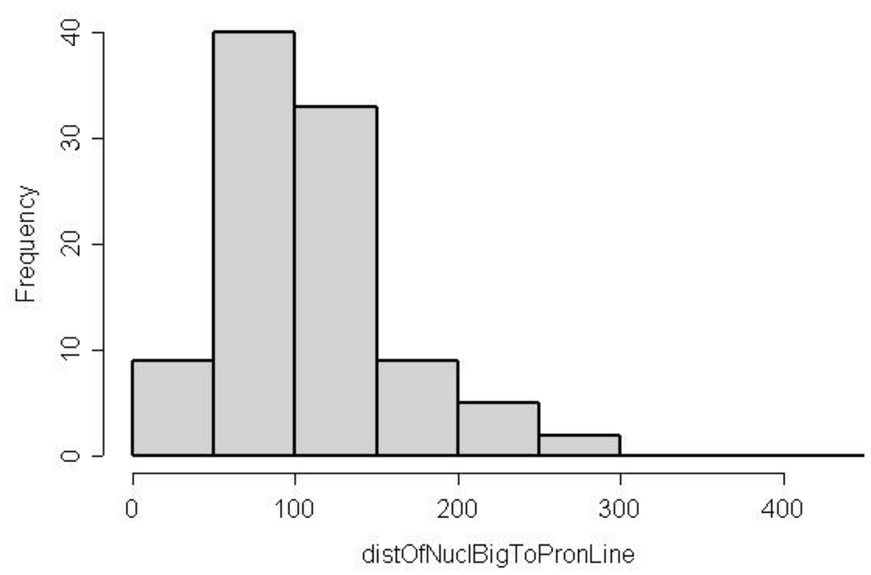

Histogram: good
Frequency
0
5
10
15
20
2
4
6
8
10
nucleolesBig

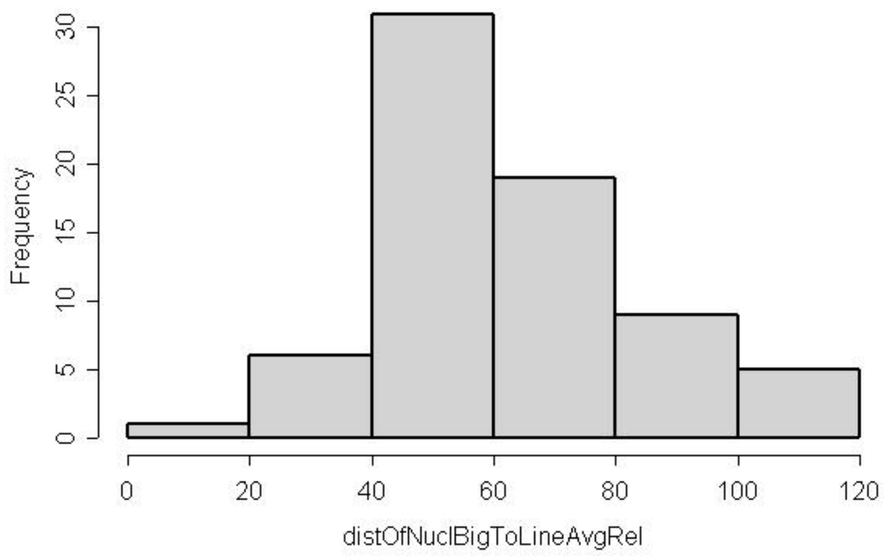

Histogram: bad
Frequency
0
5
10
15
20
25
2
4
6
8
10
nucleolesBig

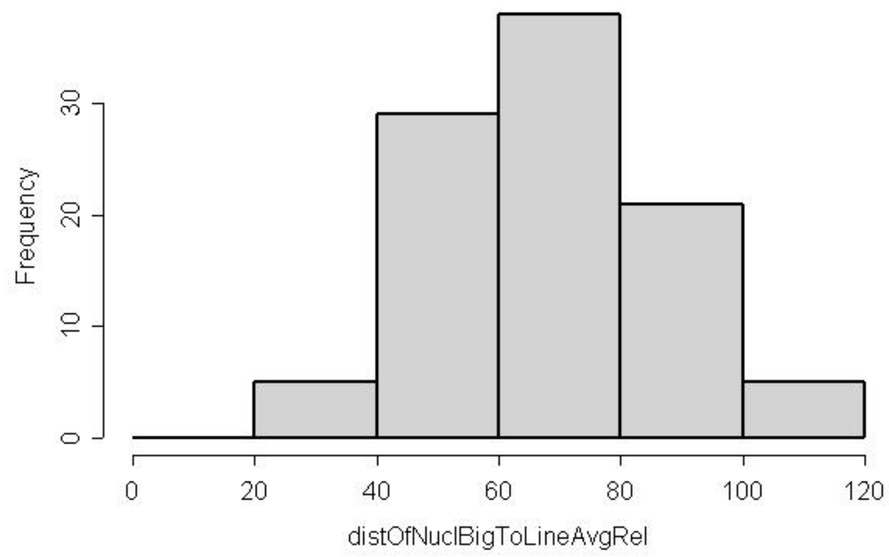

Histogram: good
Frequency
0
5
10
15
20
5
10
15
20
25
dispOfNuclBigAvg

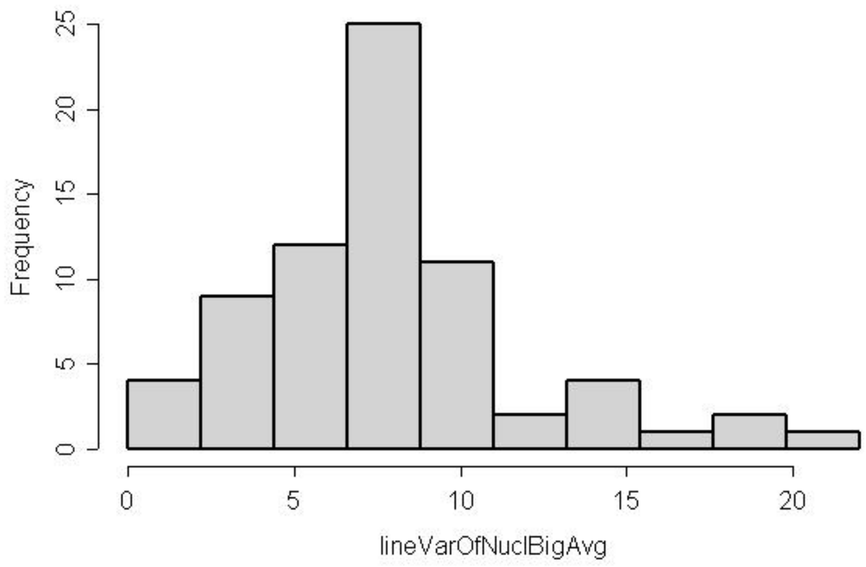

Histogram: bad
Frequency
0
5
10
15
20
25
30
5
10
15
20
25
dispOfNuclBigAvg

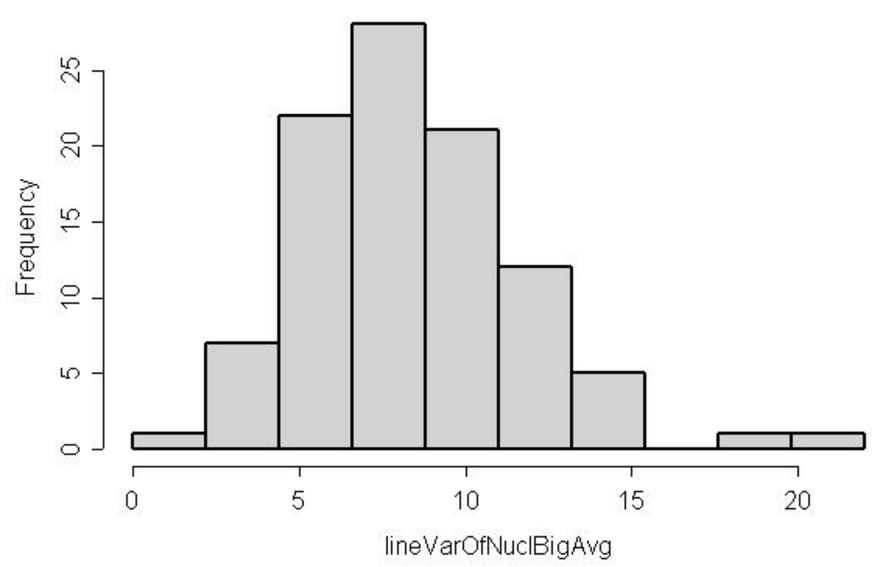

Histogram: good
Frequency
0
5
10
15
20
25
30
35
0
100
200
300
400
distOfNuclBigToPronLine

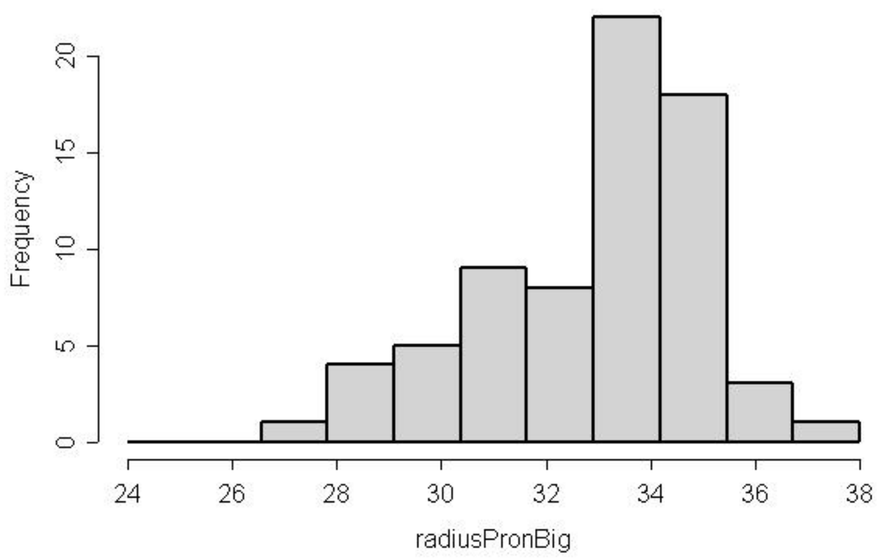

Histogram: bad
Frequency
0
10
20
30
40
0
100
200
300
400
distOfNuclBigToPronLine

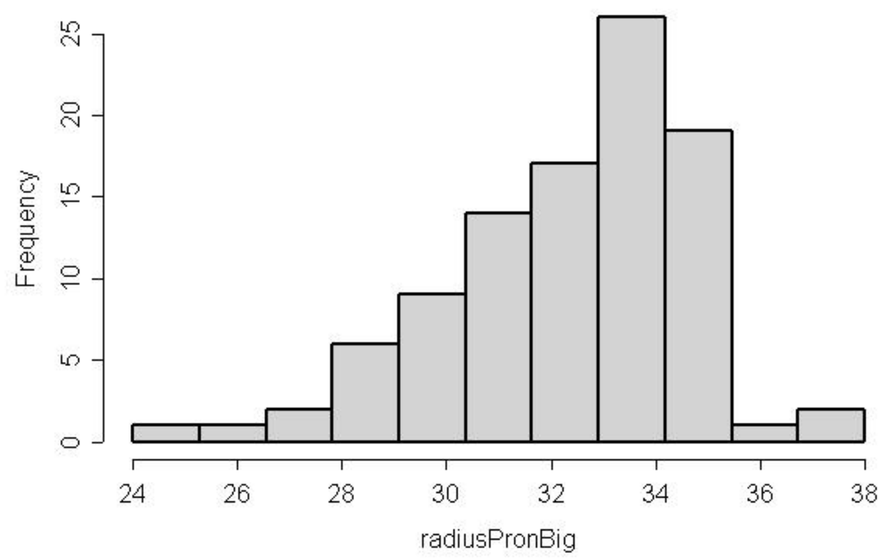

Histogram: good
Frequency
0
5
10
15
20
25
30
0
20
40
60
80
100
120
distOfNuclBigToLineAvgRel

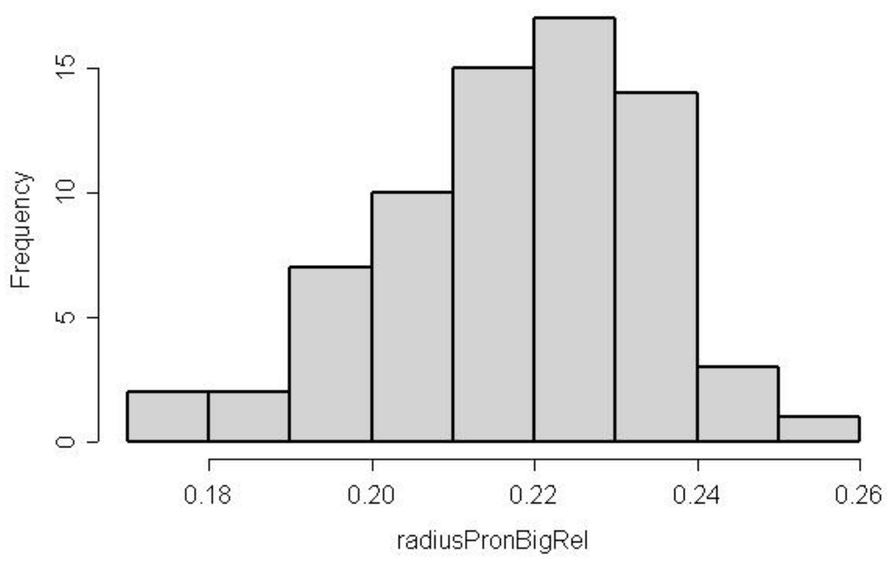

Histogram: bad
Frequency
0
10
20
30
0
20
40
60
80
100
120
distOfNuclBigToLineAvgRel

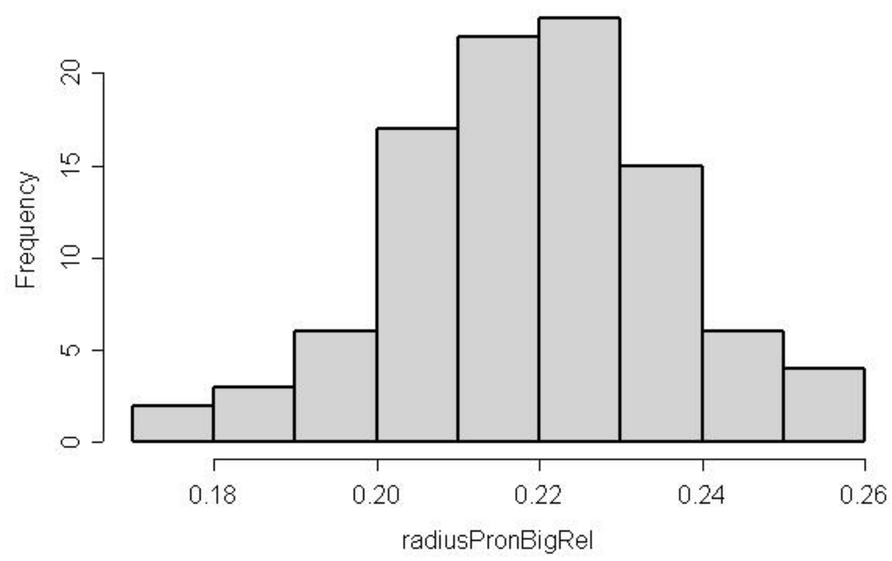

Histogram: good
Frequency
0
5
10
15
20
25
0
5
10
15
20
lineVarOfNuclBigAvg


Histogram: bad
Frequency
0
5
10
15
20
25
0
5
10
15
20
lineVarOfNuclBigAvg


Histogram: good
Frequency
0
5
10
15
20
24
26
28
30
32
34
36
38
radiusPronBig


Histogram: bad
Frequency
0
5
10
15
20
25
24
26
28
30
32
34
36
38
radiusPronBig


Histogram: good
Frequency
0
5
10
15
0.18
0.20
0.22
0.24
0.26
radiusPronBigRel

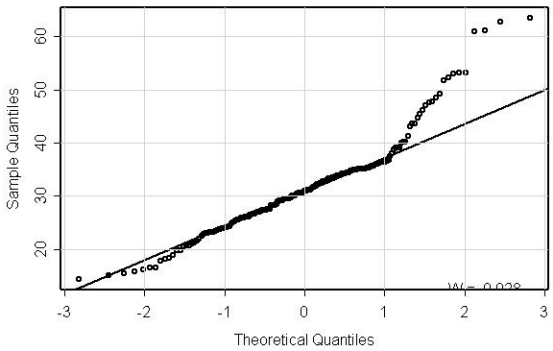

Histogram: bad
Frequency
0
5
10
15
20
0.18
0.20
0.22
0.24
0.26
radiusPronBigRel

Appendix C

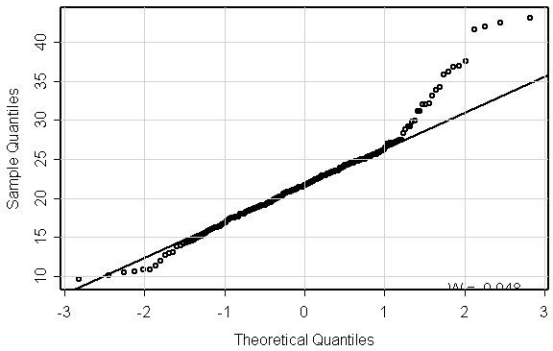
QQ-plots

advantages:	sample sizes do not need to be equal, shifts in location, in scale and changes in symmetry can be detected presence of outliers can be detected
disadvantages:	need to be compared to another distribution

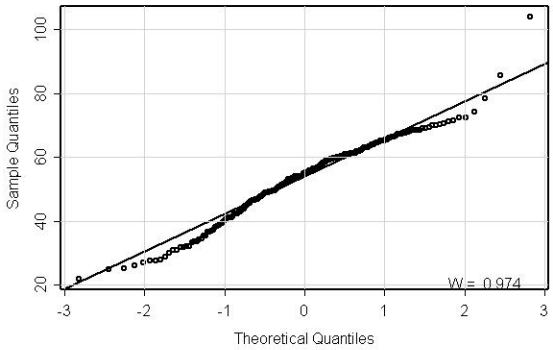
Q-Q Plot posOfPronSum



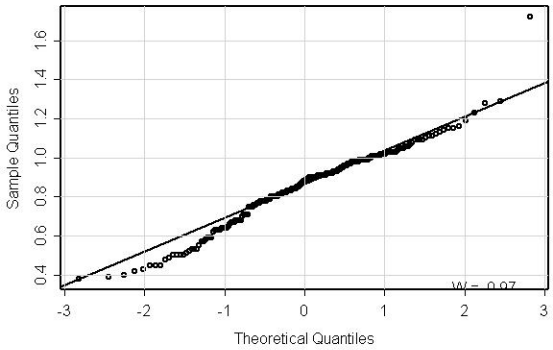
Q-Q Plot posOfPronRel



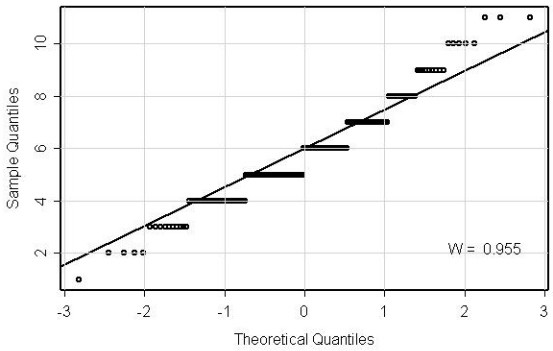
Q-Q Plot proxOfPronCent



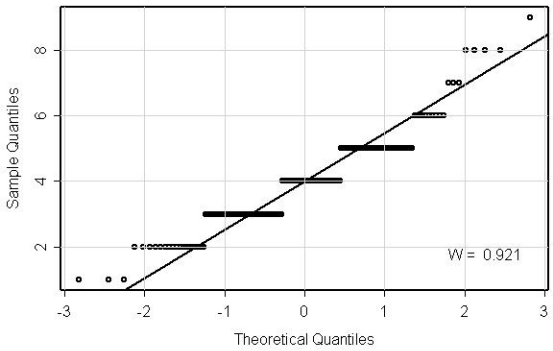
Q-Q Plot proxOfPronRel



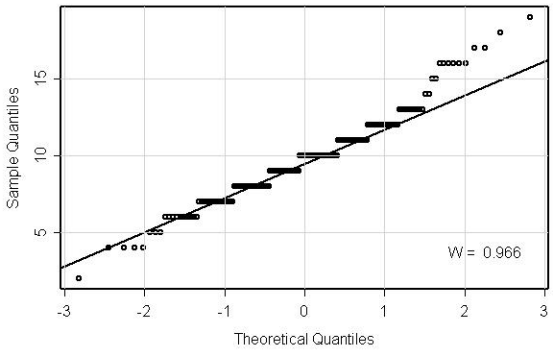
Q-Q Plot nucleoles1



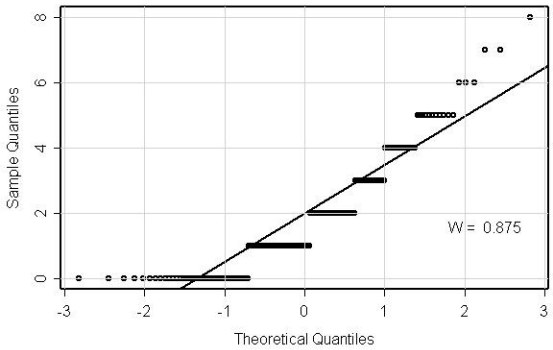
Q-Q Plot nucleoles2



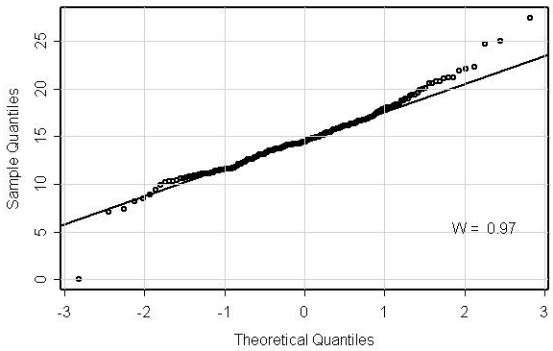
Q-Q Plot sumOfNucl



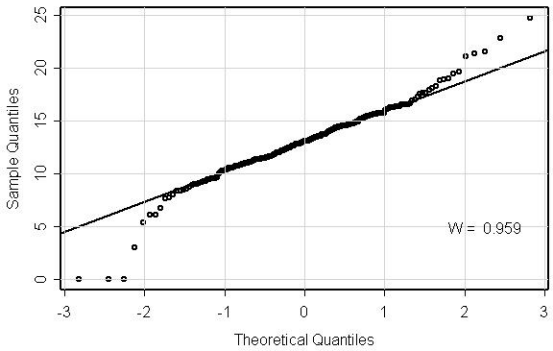
Q-Q Plot diffOfNucl



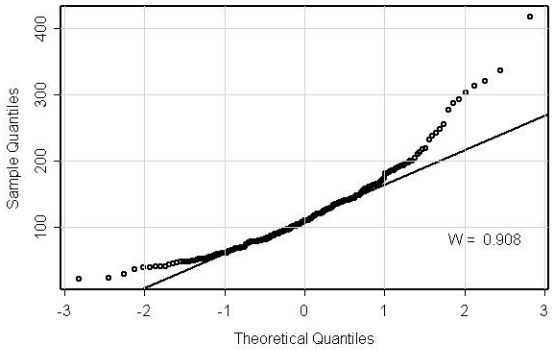
Q-Q Plot dispOfNucl1Avg



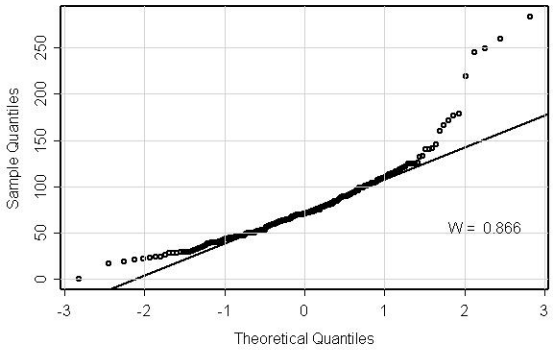
Q-Q Plot dispOfNucl2Avg



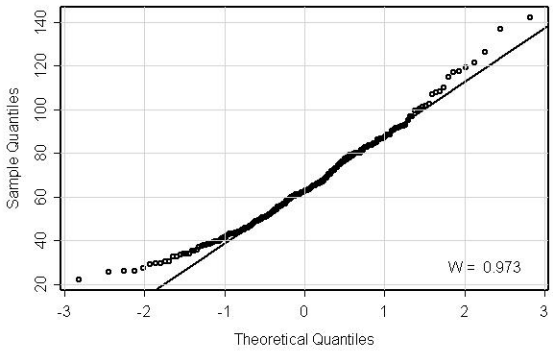
Q-Q Plot distOfNucl1ToPronLine



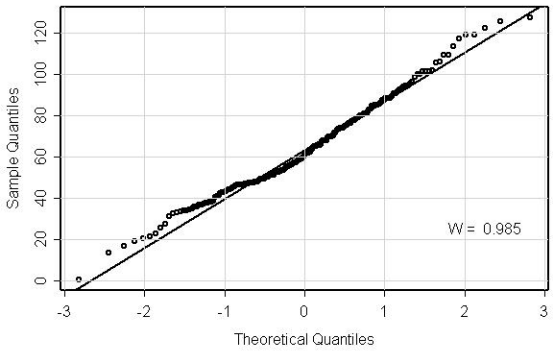
Q-Q Plot distOfNucl2ToPronLine



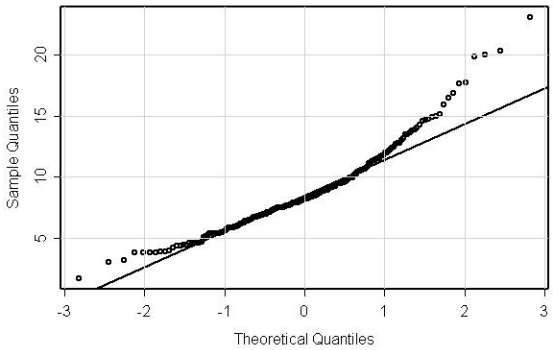
Q-Q Plot distOfNucl1ToLineAvgRel



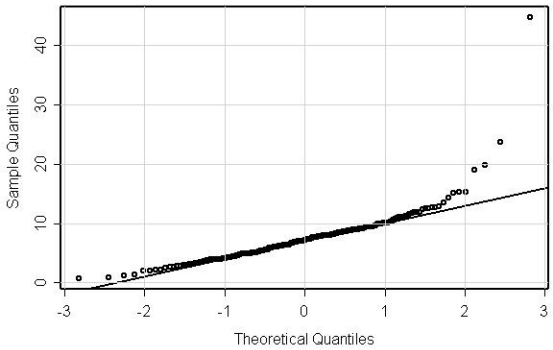
Q-Q Plot distOfNucl2ToLineAvgRel



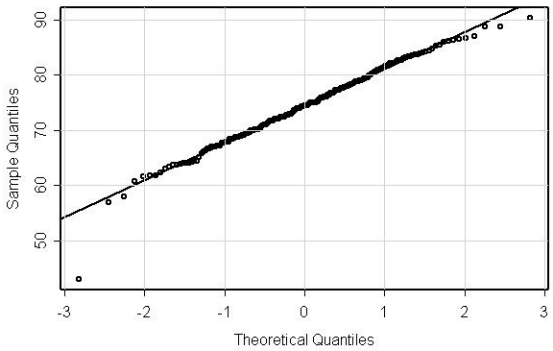
Q-Q Plot lineVarOfNucl1Avg



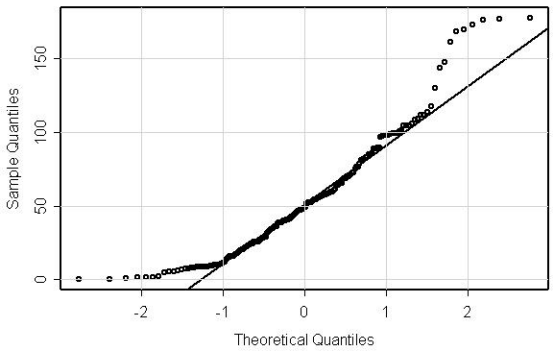
Q-Q Plot lineVarOfNucl2Avg



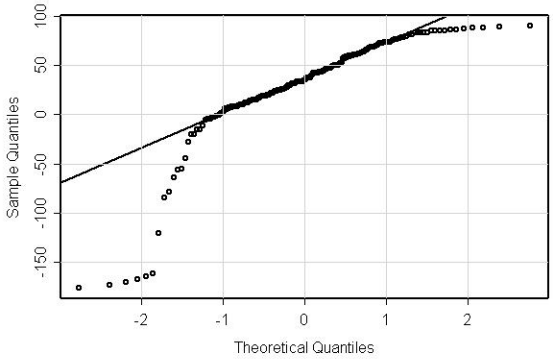
Q-Q Plot corticalReaction



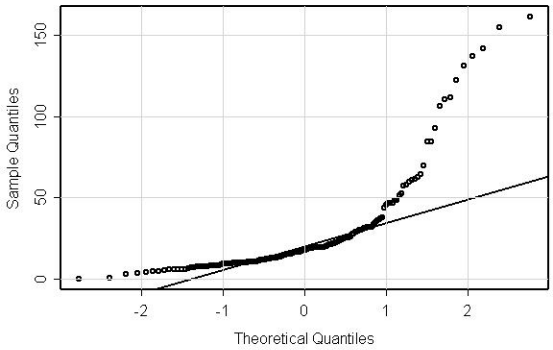
Q-Q Plot alphaMin



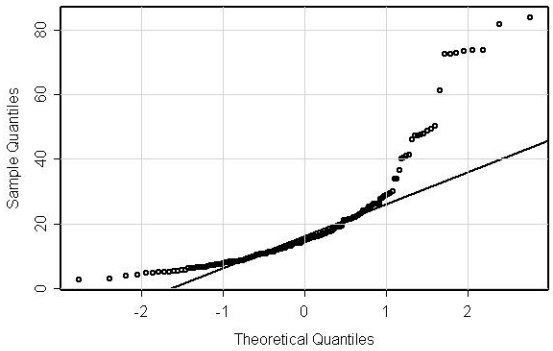
Q-Q Plot alphaMax



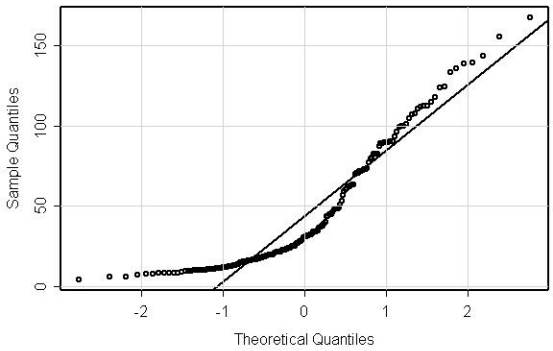
Q-Q Plot alphaBetween



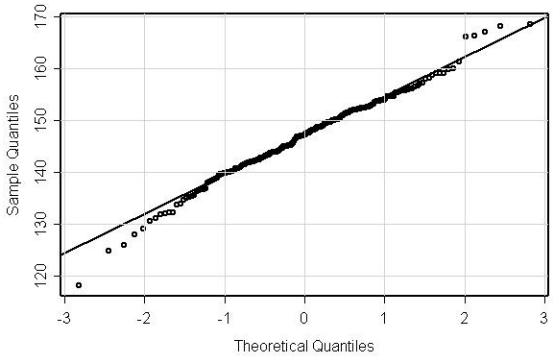
Q-Q Plot alpha3D1



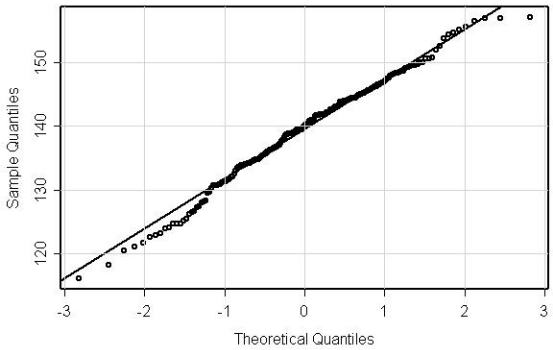
Q-Q Plot alpha3D2



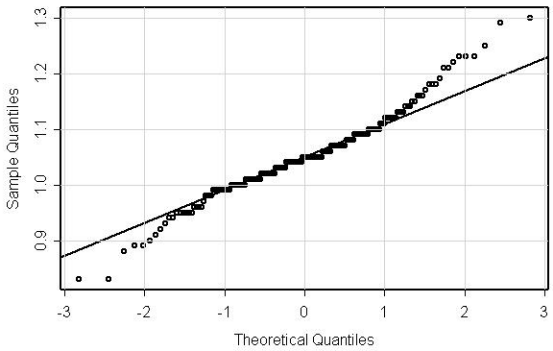
Q-Q Plot OolemmaA



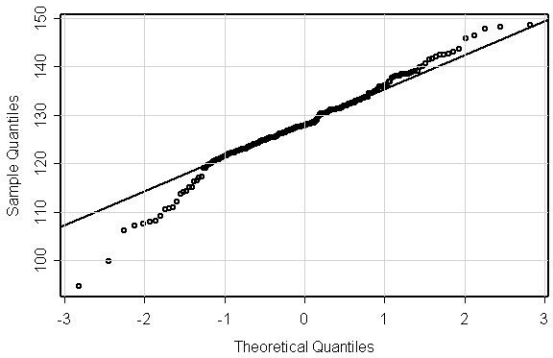
Q-Q Plot OolemmaB



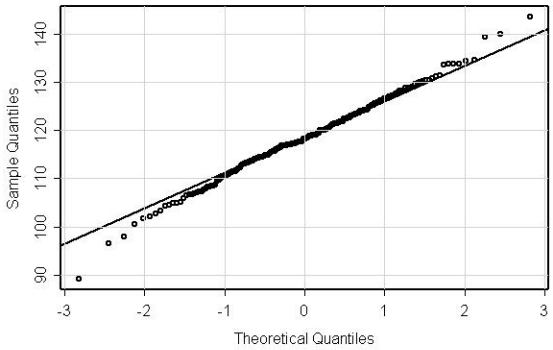
Q-Q Plot OolemmaRatio



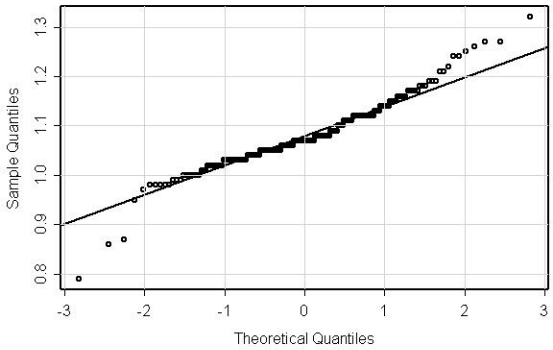
Q-Q Plot CorticalA



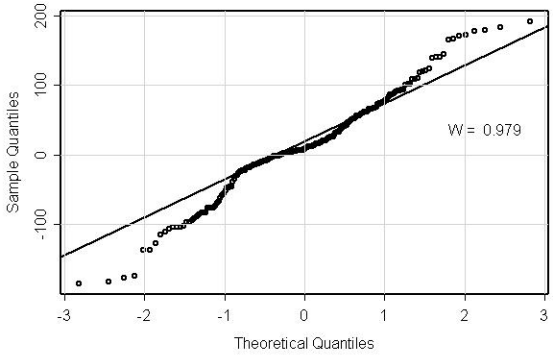
Q-Q Plot CorticalB



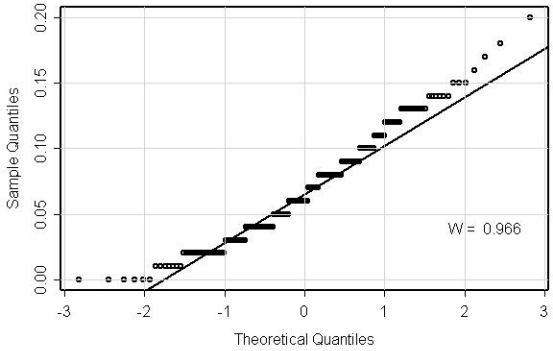
Q-Q Plot CorticalRatio



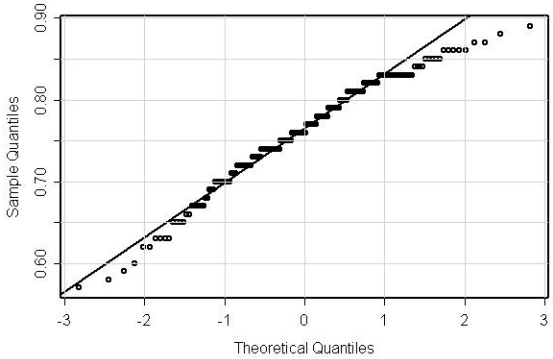
Q-Q Plot Ellipse.angle



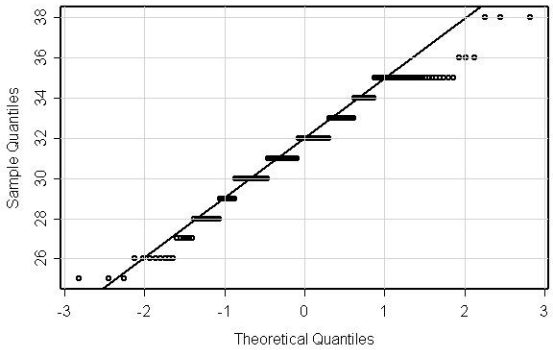
Q-Q Plot ellDistCentersRel



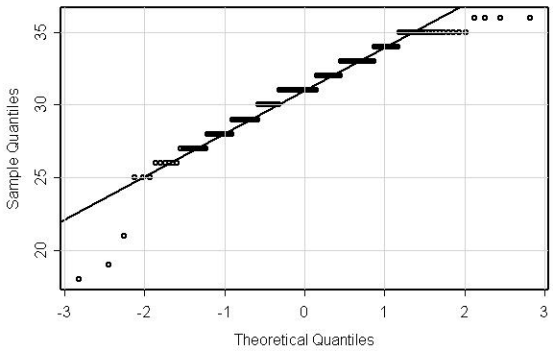
Q-Q Plot ellDistBetweenRel



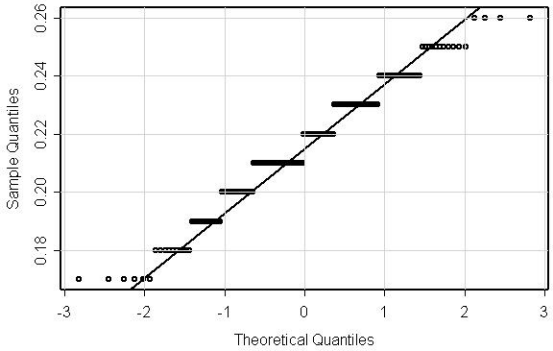
Q-Q Plot radiusPron1



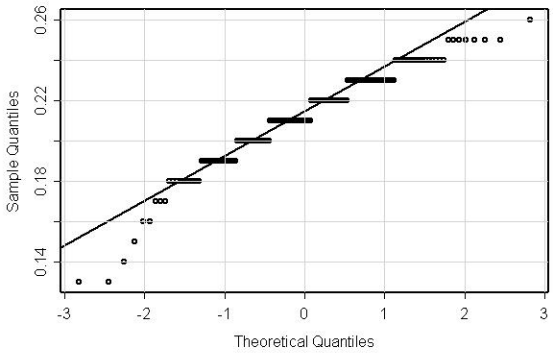
Q-Q Plot radiusPron2



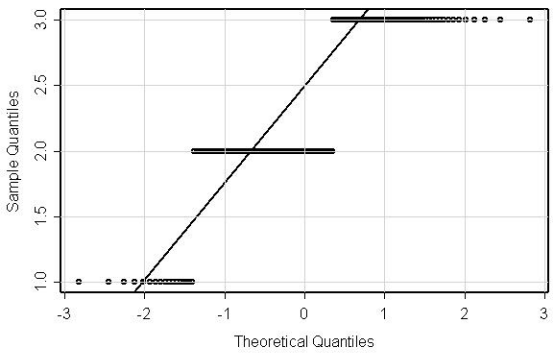
Q-Q Plot radiusPron1Rel



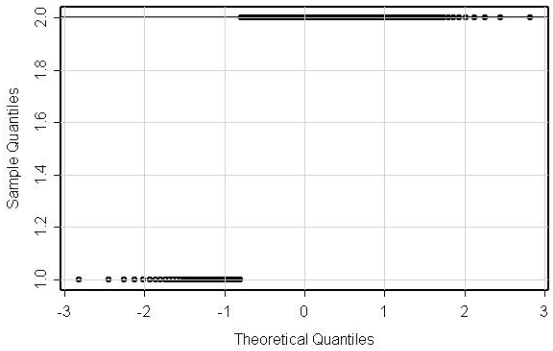
Q-Q Plot radiusPron2Rel



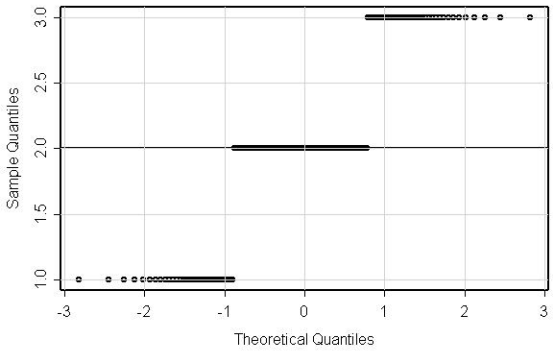
Q-Q Plot ZS1.Centering



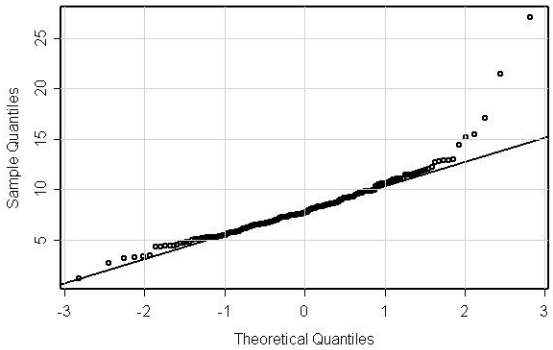
Q-Q Plot ZS2.Proximity



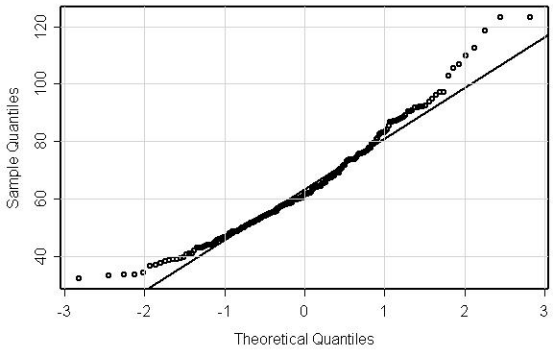
Q-Q Plot ZS.3.Nb.of.nucl



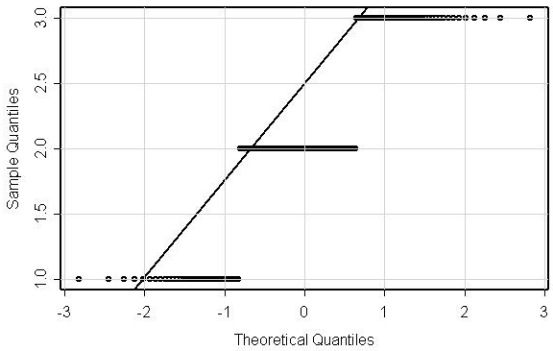
Q-Q Plot ZS4.Line.dispersion



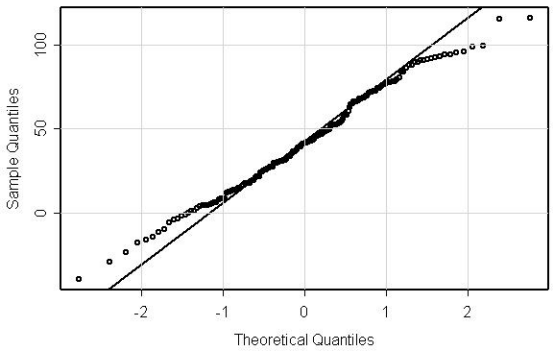
Q-Q Plot ZS.4bis.Line.position



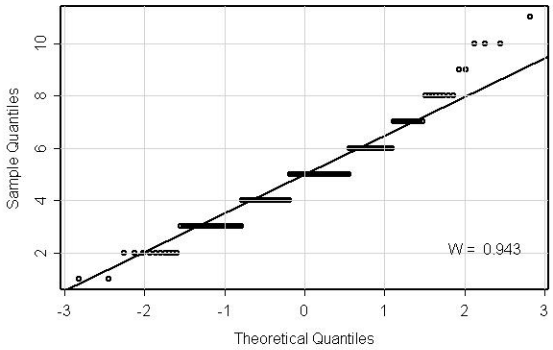
Q-Q Plot ZS5.Cort.react



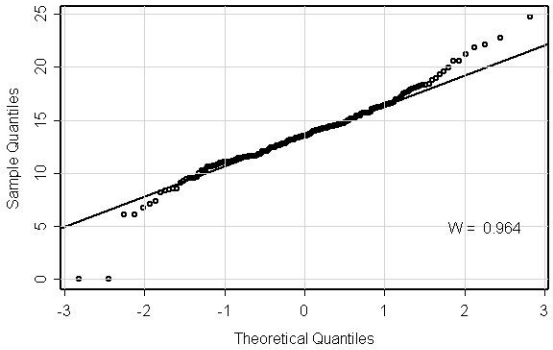
Q-Q Plot ZS6.GP.avg.if.90



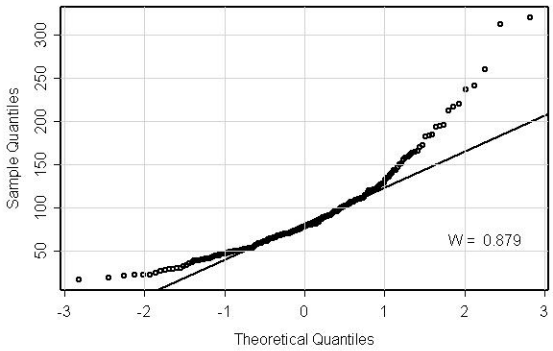
Q-Q Plot nucleolesTiny



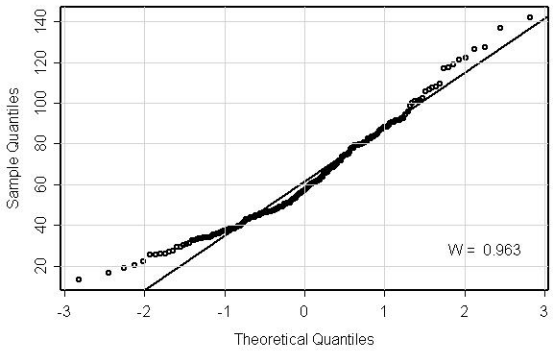
Q-Q Plot dispOfNuclTinyAvg



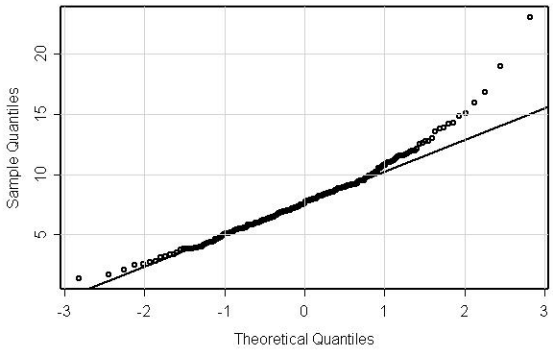
Q-Q Plot distOfNuclTinyToPronLine



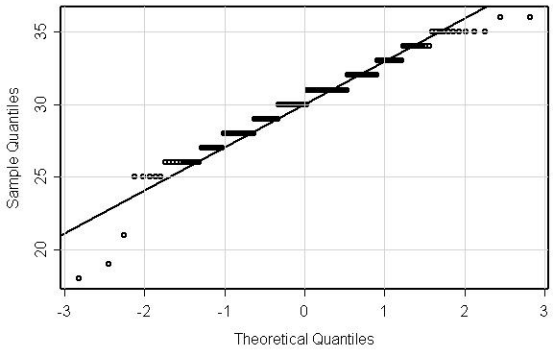
Q-Q Plot distOfNuclTinyToLineAvgRel



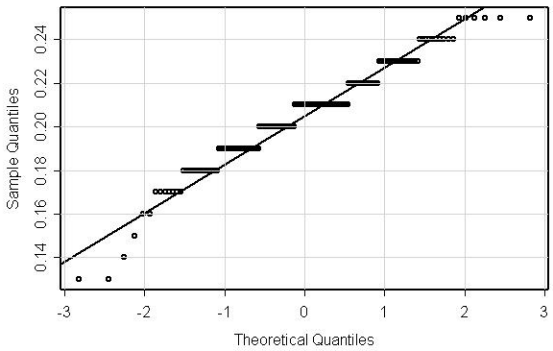
Q-Q Plot lineVarOfNuclTinyAvg



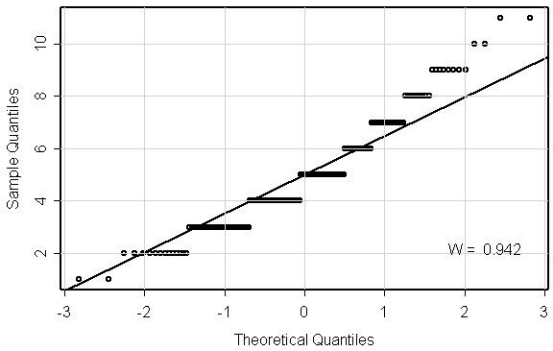
Q-Q Plot radiusPronTiny



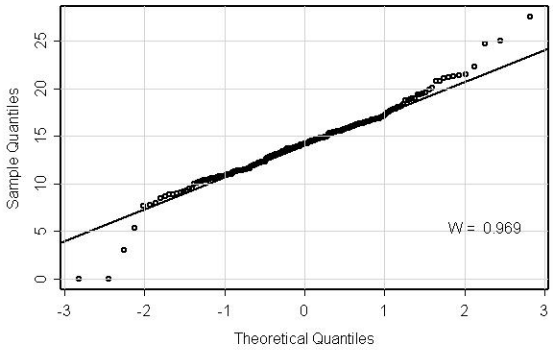
Normal Q-Q Plot radiusPronTinyRel



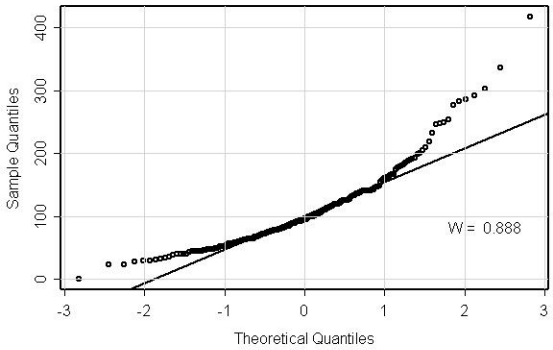
Normal Q-Q Plot nucleolesBig



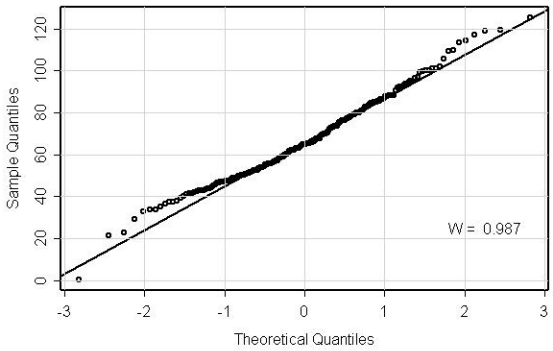
Normal Q-Q Plot dispOfNuclBigAvg



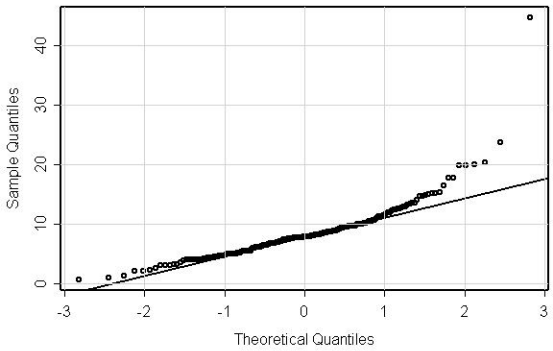
Normal Q-Q Plot distOfNuclBigToPronLine



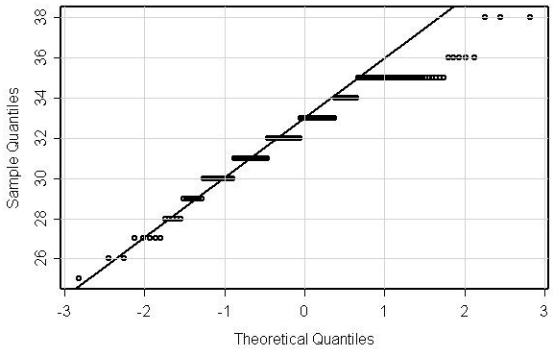
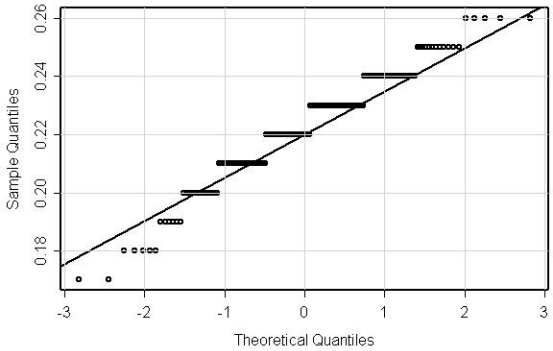
Normal Q-Q Plot distOfNuclBigToLineAvgRel



Normal Q-Q Plot lineVarOfNuclBigAvg



Normal Q-Q Plot radiusPronBig



Normal Q-Q Plot radiusPronBigRel