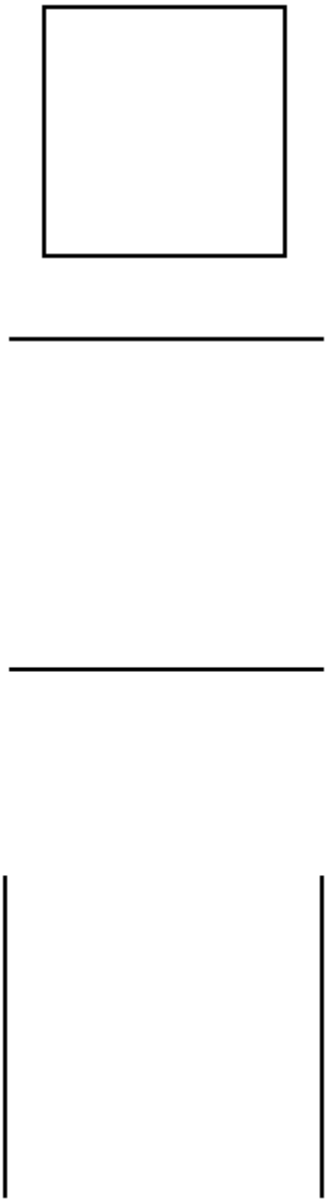
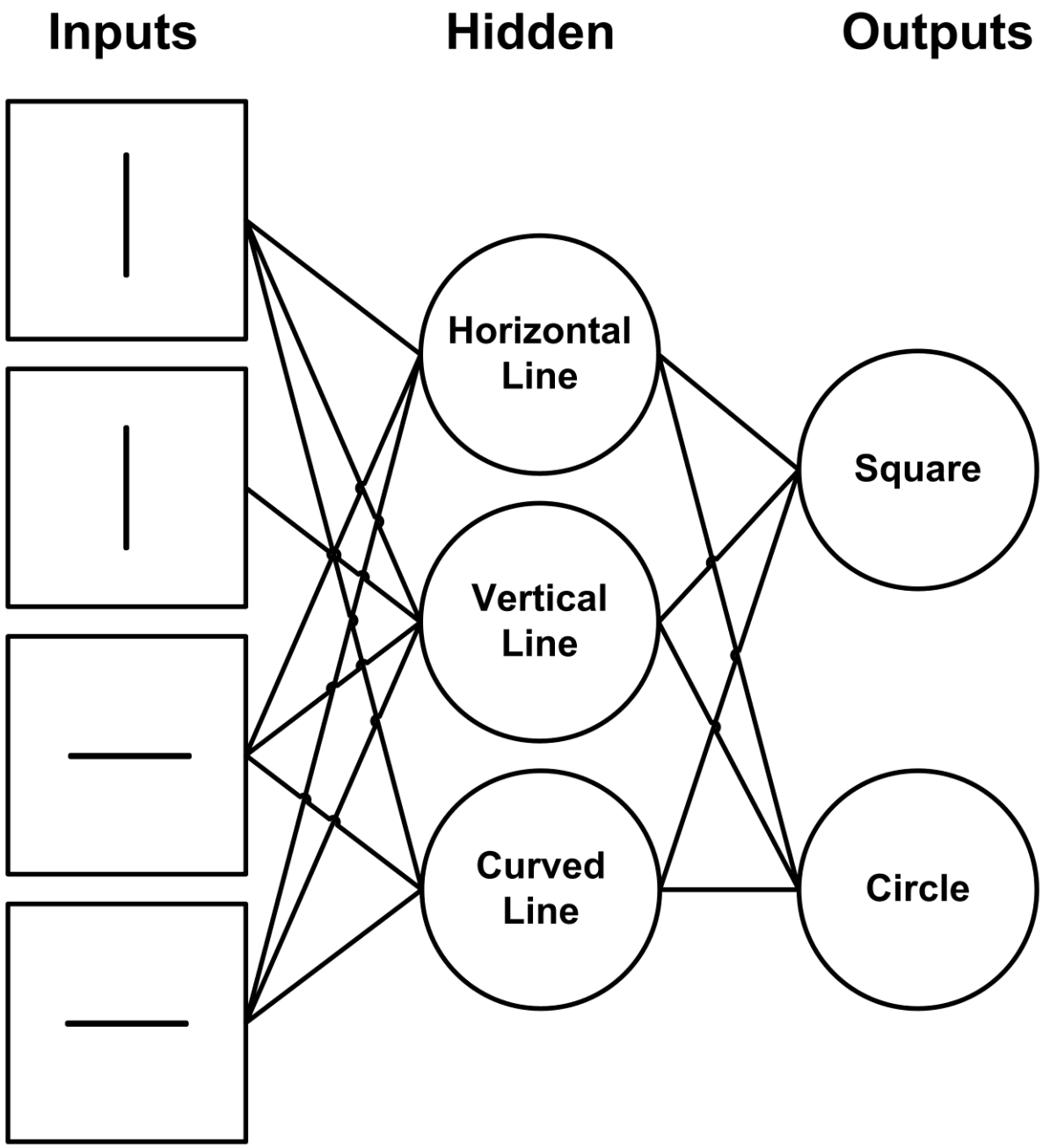
Chapter 1: Getting Started with Deep Learning



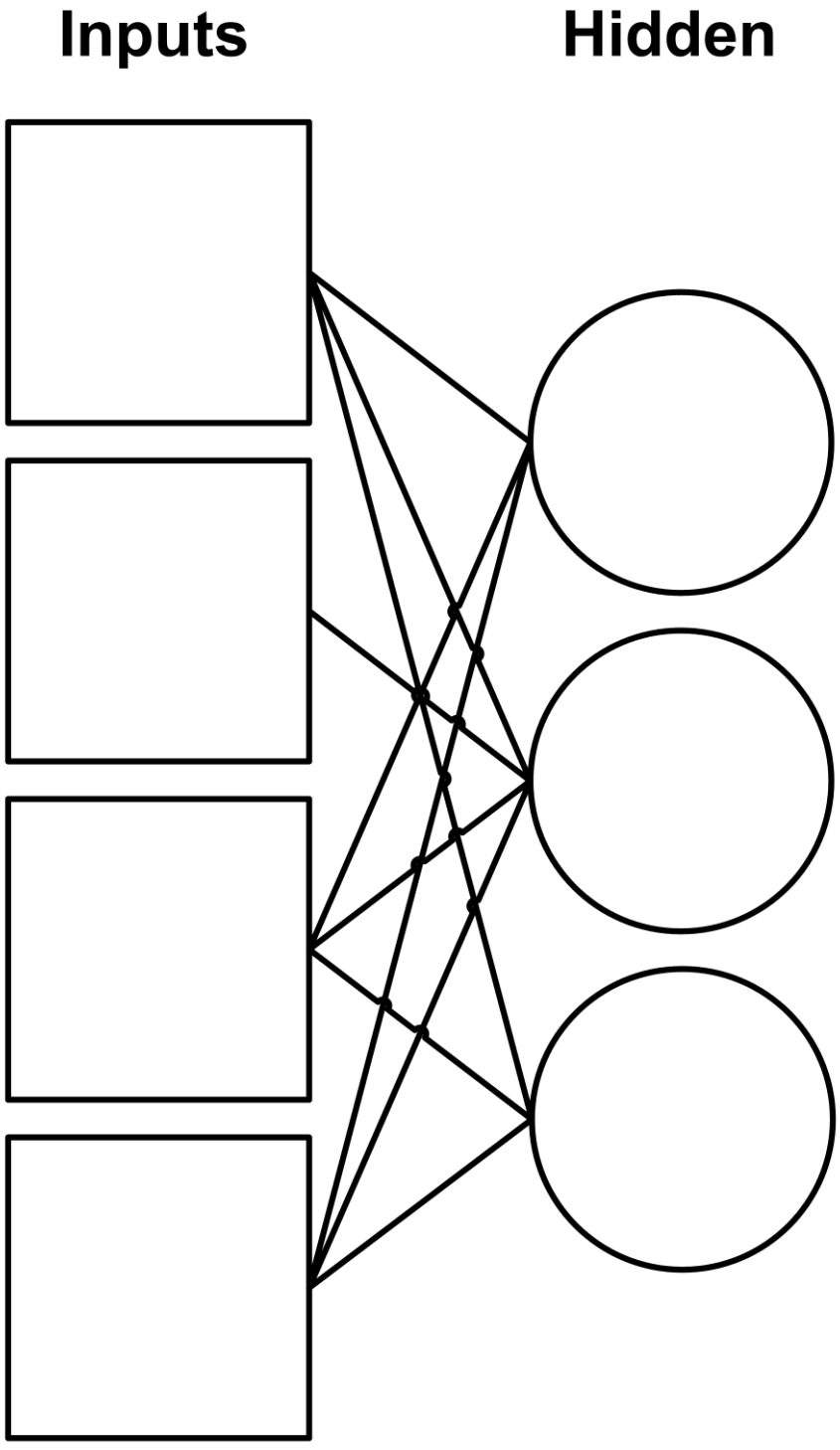


$$\sigma(x) = \frac{1}{1+e^{-x}}$$

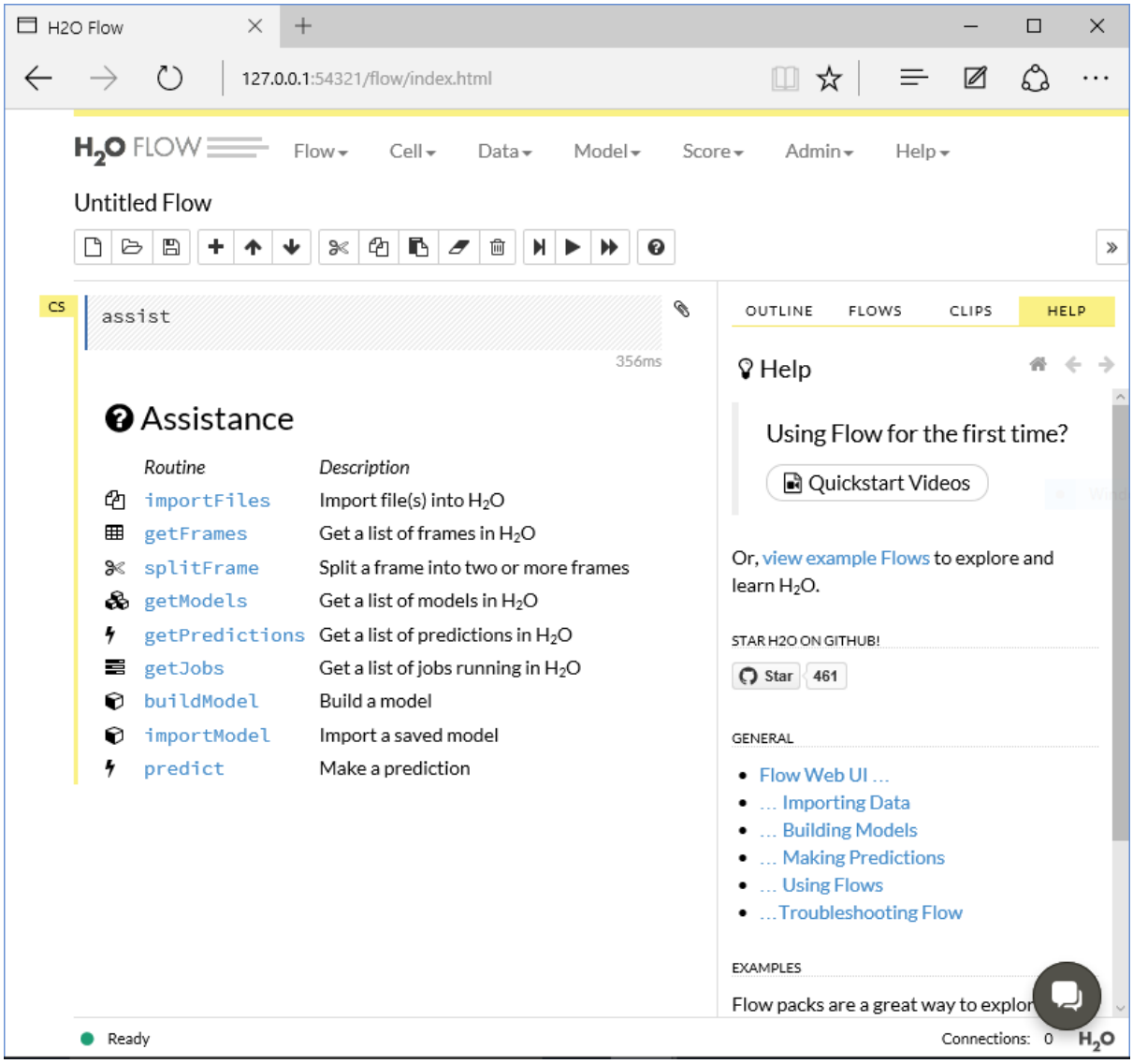
$$f(x) = \tanh(x)$$

$$f(x) = \exp\left(-\frac{\|x-c\|^2}{2\sigma^2}\right)$$

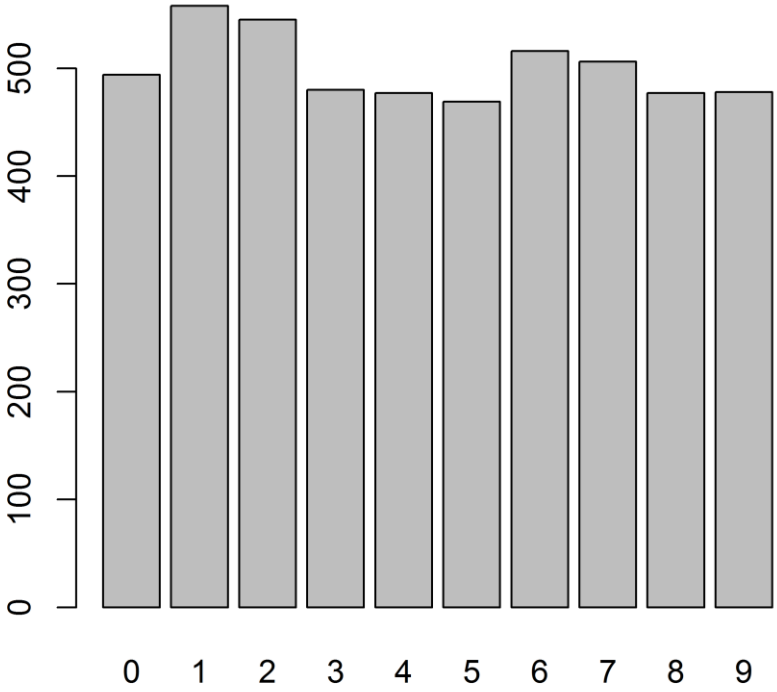
$$Y_i = \frac{e^{w_i^T h}}{\sum_i^k e^{w_i^T h}}$$

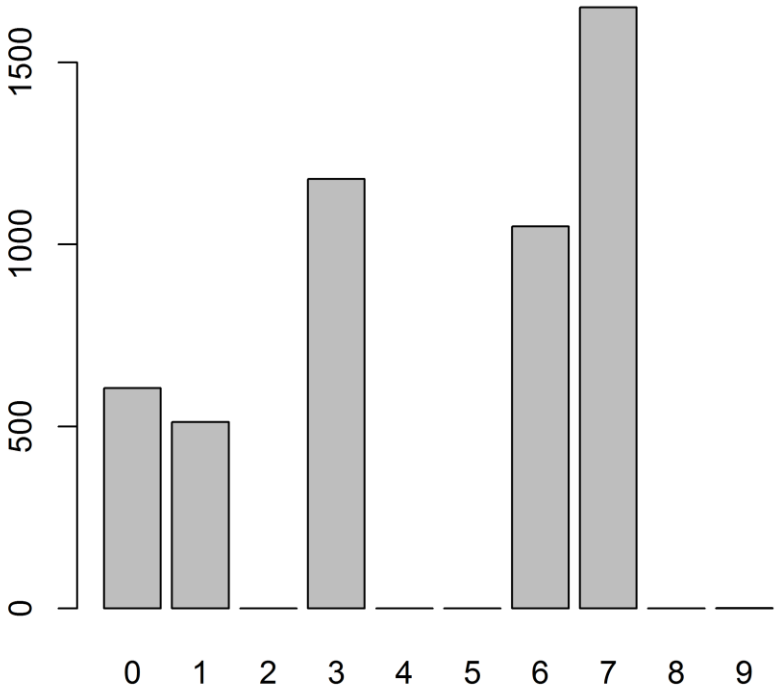

Inputs
Hidden

```
File  Edit  Options  Buffers  Tools  iESS  Complete  In/Out  Signals  Help

R version 3.2.3 (2015-12-10) -- "Wooden Christmas-Tree"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> > options(chmhelp=FALSE, help_type="text")
> options(STERM='iESS', str.dendrogram.last="'", editor='emacsclient.exe', show.
error.locations=TRUE)
> library(checkpoint)

checkpoint: Part of the Reproducible R Toolkit from Revolution Analytics
http://projects.revolutionanalytics.com/rrt/
> checkpoint("2016-02-20", R.version = "3.2.3")
Can I create directory ~/.checkpoint for internal checkpoint use?(y/n)
y
Scanning for packages used in this project
  |======================================================================| 100%
- Discovered 7 packages
Installing packages used in this project
 - Installing 'caret'
also installing the dependencies 'colorspace', 'minqa', 'nloptr', 'RcppEigen', '
RColorBrewer', 'dichromat', 'munsell', 'labeling', 'Matrix', 'lme4', 'SparseM',
'MatrixModels', 'stringi', 'magrittr', 'digest', 'gtable', 'MASS', 'scales', 'mg
cv', 'nnet', 'pbkrtest', 'quantreg', 'codetools', 'iterators', 'Rcpp', 'stringr'
, 'lattice', 'ggplot2', 'car', 'foreach', 'plyr', 'nlme', 'reshape2'

package 'colorspace' successfully unpacked and MD5 sums checked
[ output cut ]
package 'caret' successfully unpacked and MD5 sums checked
 - Installing 'e1071'
also installing the dependency 'class'

package 'class' successfully unpacked and MD5 sums checked
package 'e1071' successfully unpacked and MD5 sums checked
 - Installing 'jsonlite'
package 'jsonlite' successfully unpacked and MD5 sums checked
 - Previously installed 'MASS'
 - Installing 'RCurl'
also installing the dependency 'bitops'

package 'bitops' successfully unpacked and MD5 sums checked
package 'RCurl' successfully unpacked and MD5 sums checked
 - Installing 'statmod'
package 'statmod' successfully unpacked and MD5 sums checked
checkpoint process complete
---
> 
1\**-  *R*      All (54,2)     (iESS [R db -]: run company ElDoc)
```


H2O Flow
127.0.0.1:54321/flow/index.html
H2O FLOW
Flow
Cell
Data
Model
Score
Admin
Help
Untitled Flow
CS
assist
356ms
Assistance
Routine
Description
importFiles
Import file(s) into H2O
getFrames
Get a list of frames in H2O
splitFrame
Split a frame into two or more frames
getModels
Get a list of models in H2O
getPredictions
Get a list of predictions in H2O
getJobs
Get a list of jobs running in H2O
buildModel
Build a model
importModel
Import a saved model
predict
Make a prediction
OUTLINE
FLOWS
CLIPS
HELP
Help
Using Flow for the first time?
Quickstart Videos
Or, view example Flows to explore and learn H2O.
STAR H2O ON GITHUB!
Star
461
GENERAL
Flow Web UI ...
... Importing Data
... Building Models
... Making Predictions
... Using Flows
...Troubleshooting Flow
EXAMPLES
Flow packs are a great way to explor
Ready
Connections: 0
H2O

Chapter 2: Training a Prediction Model





$$Sensitivity = \frac{TP}{TP + FN}$$

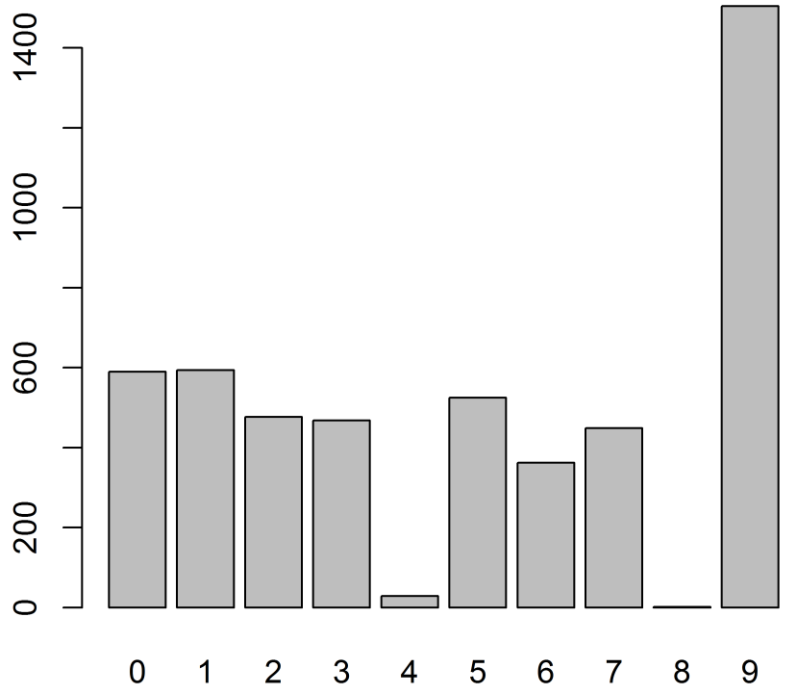
$$Specificity = \frac{TN}{TN + FP}$$

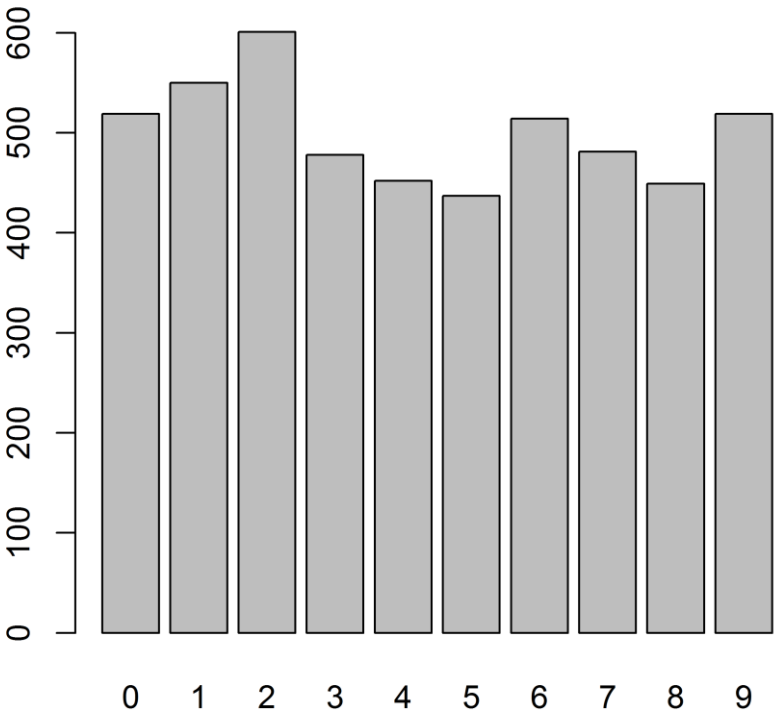
$$Positive\ Predictive\ Value\ (PPV) = \frac{TP}{TP + FP}$$

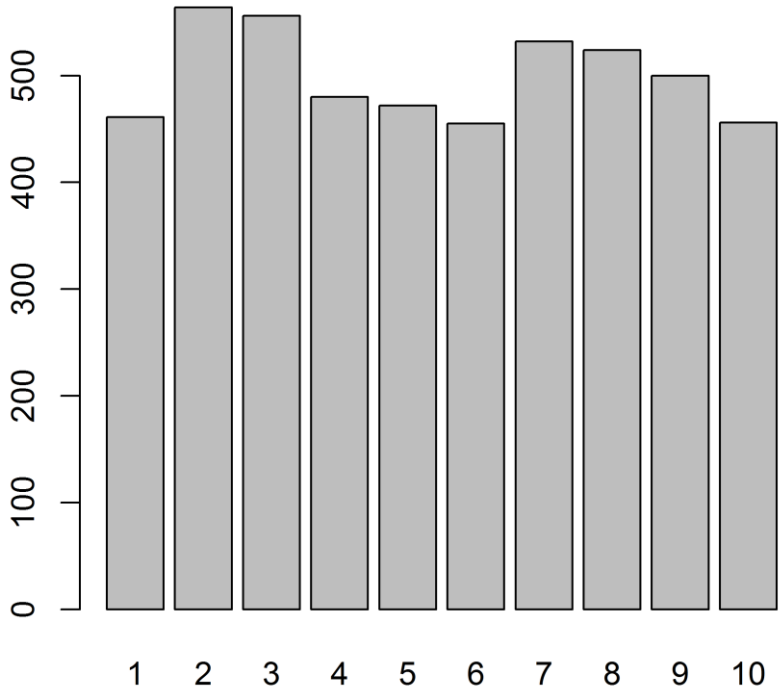
$$Negative\ Predictive\ Value\ (NPV) = \frac{TN}{FN + TN}$$

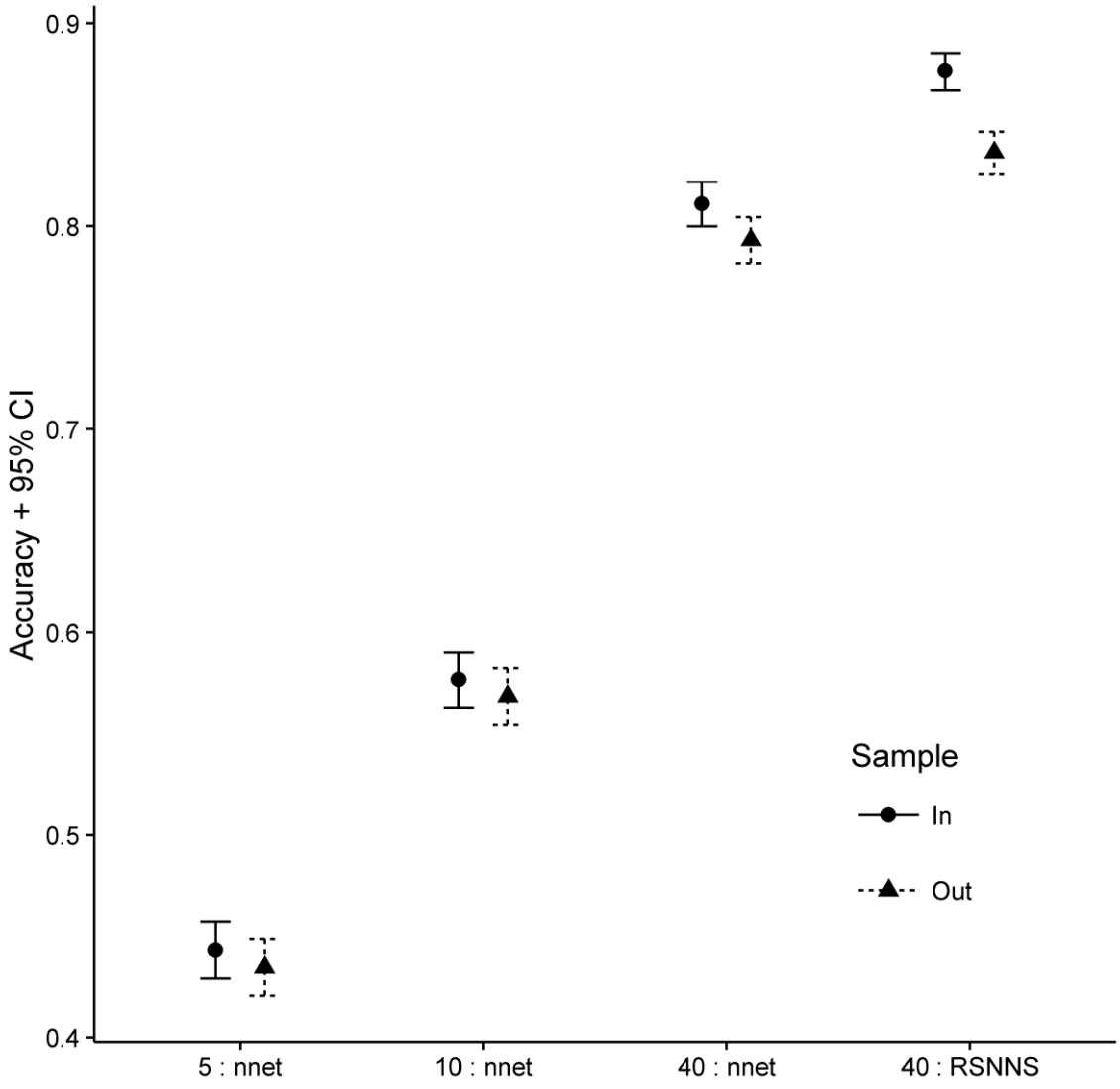
$$Detection\ Rate = \frac{TP}{TP + FN + FP + TN}$$

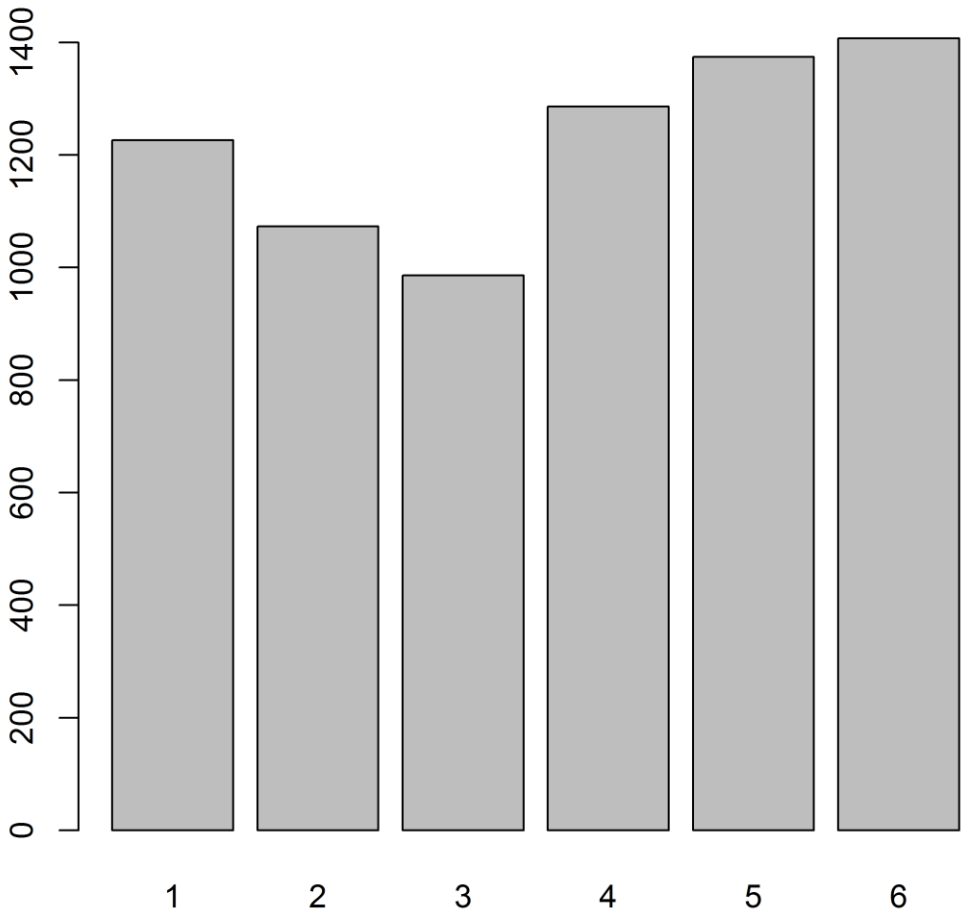
$$Detection\ Prevalence = \frac{TP + FP}{TP + FN + FP + TN}$$



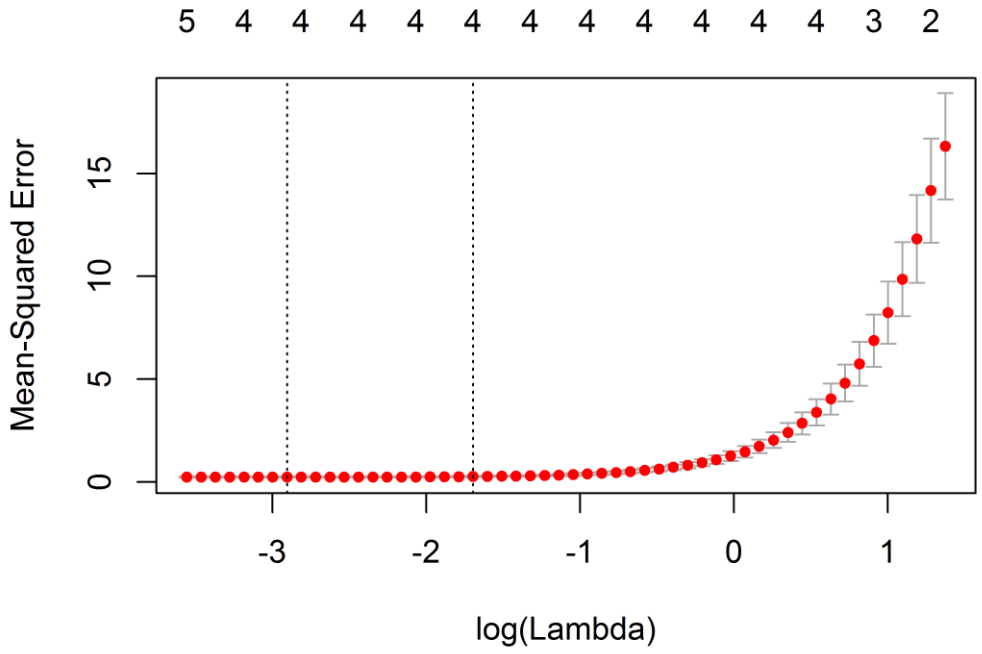

0
100
200
300
400
500
600
0 1 2 3 4 5 6 7 8 9

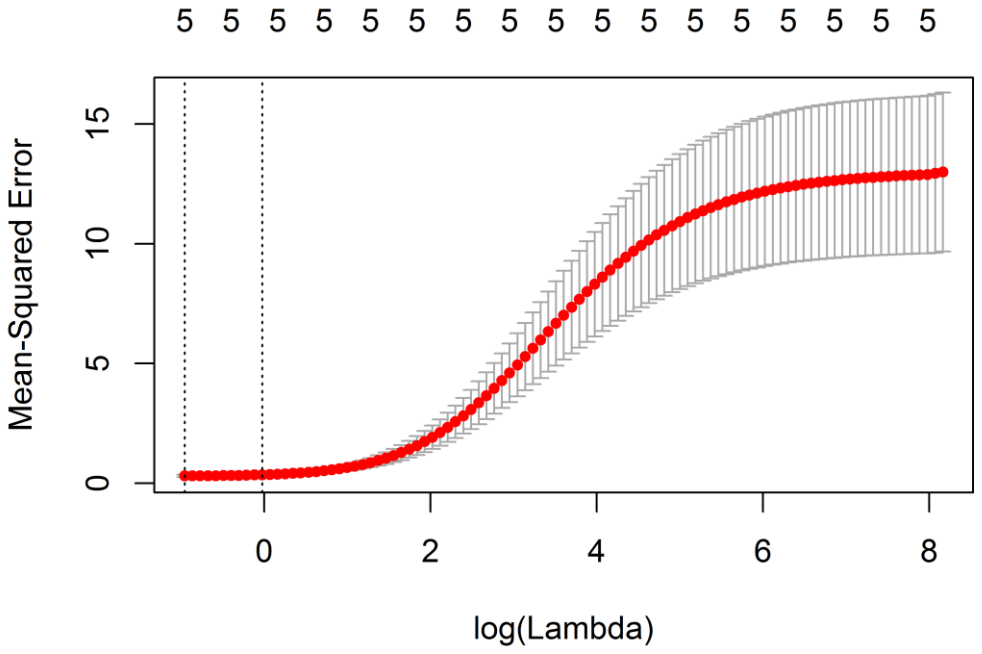

0
100
200
300
400
500
1 2 3 4 5 6 7 8 9 10

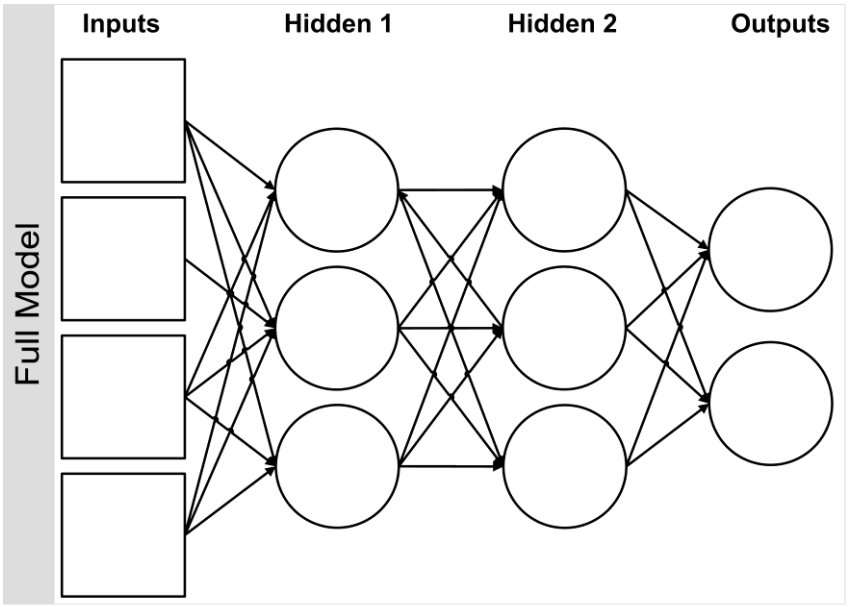

0.9
0.8
0.7
0.6
0.5
0.4
Accuracy + 95% CI
5 : nnet
10 : nnet
40 : nnet
40 : RSNNS
Sample
In
Out

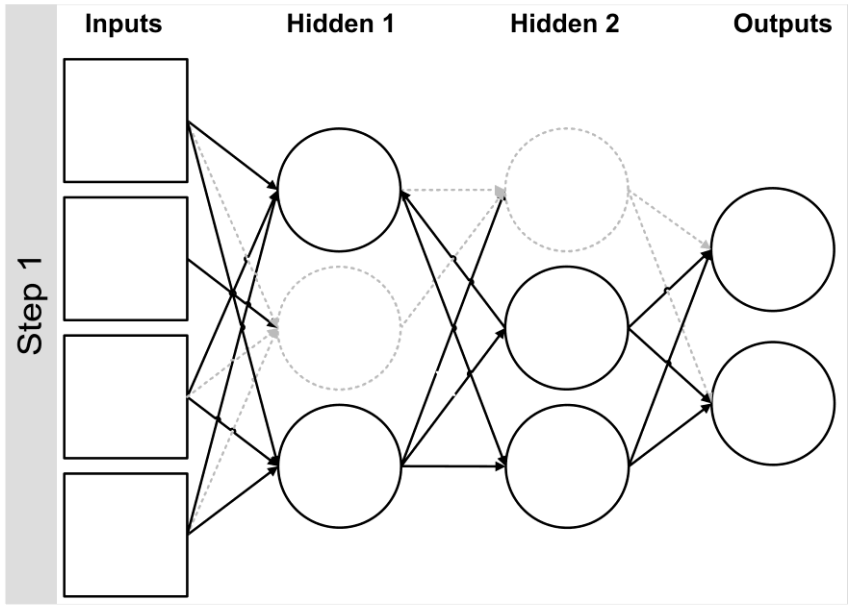

0
200
400
600
800
1000
1200
1400
1
2
3
4
5
6

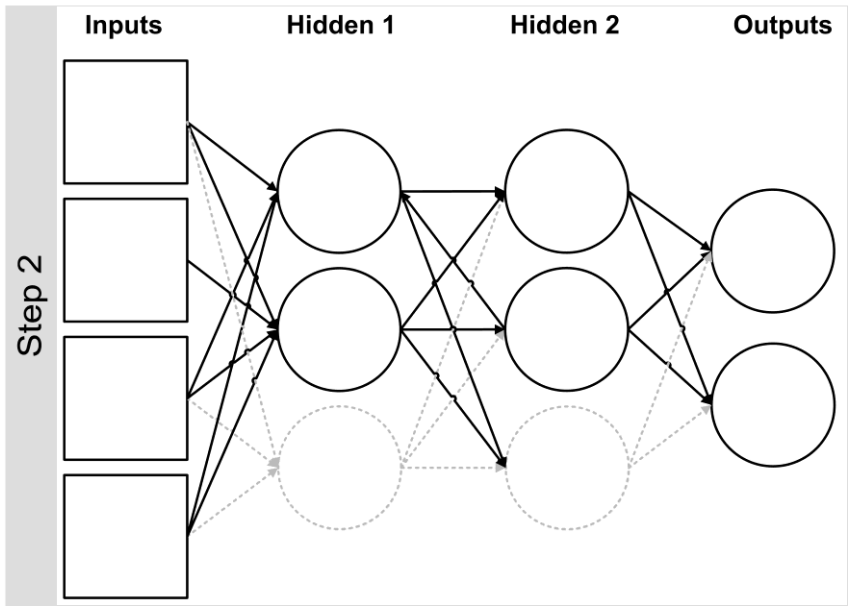
Chapter 3: Preventing Overfitting



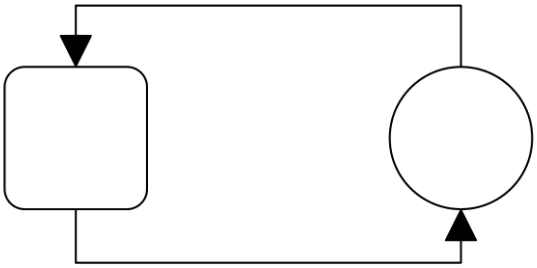



Full Model
Inputs
Hidden 1
Hidden 2
Outputs


Step 1
Inputs
Hidden 1
Hidden 2
Outputs


Step 2
Inputs
Hidden 1
Hidden 2
Outputs

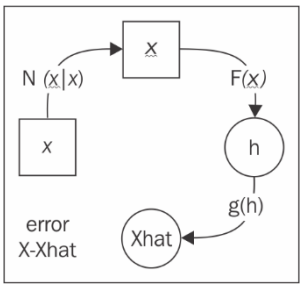
Chapter 4: Identifying Anomalous Data

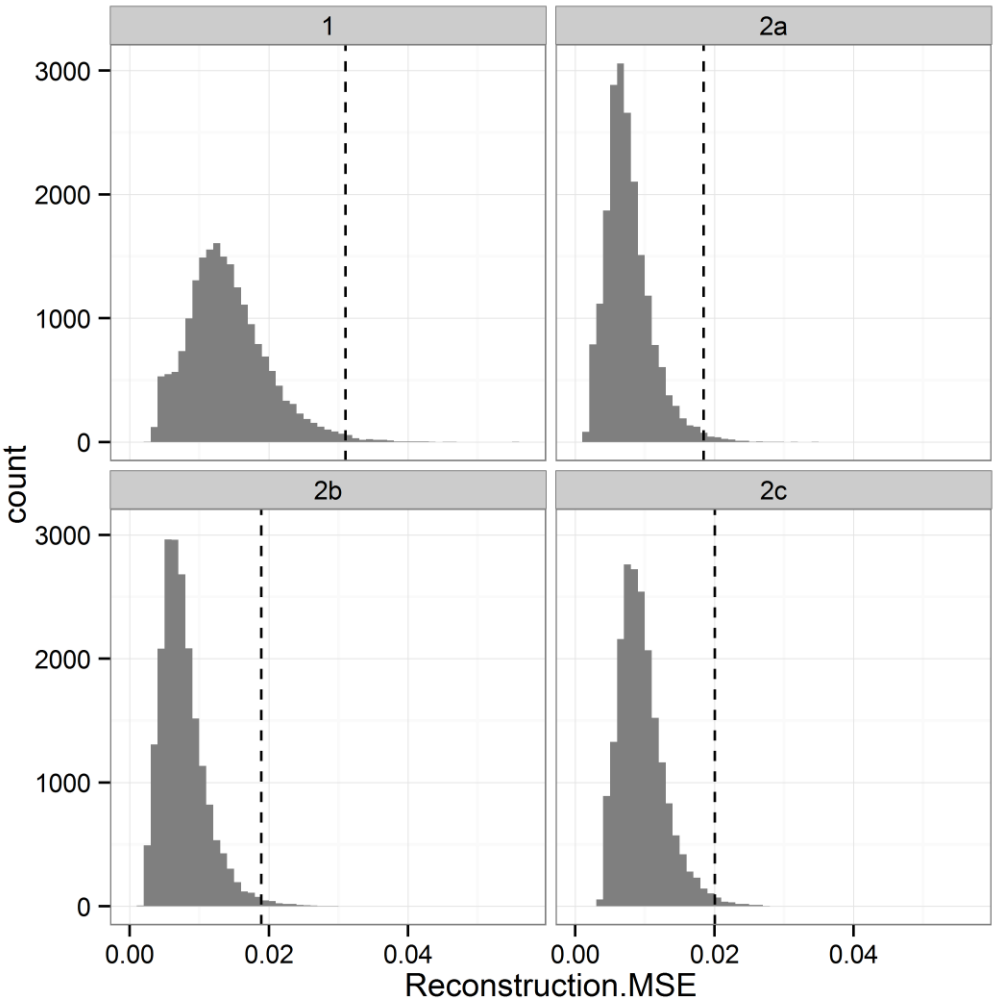


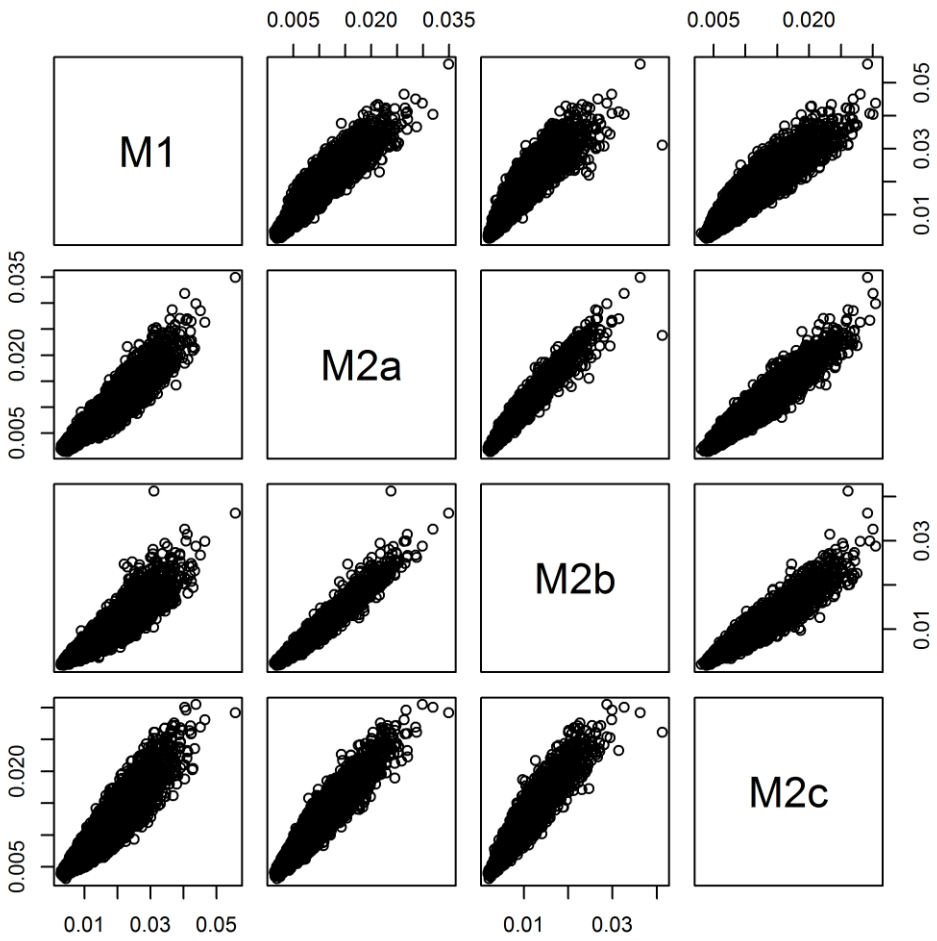
$$\underset{\sim}{x}$$

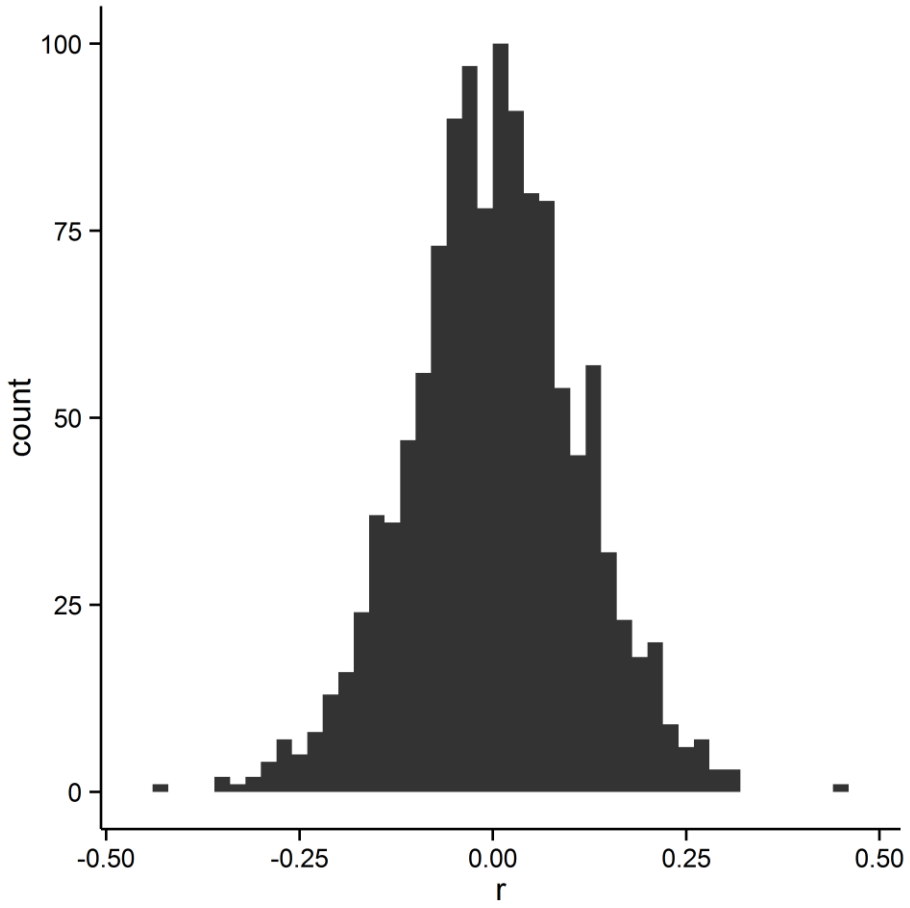
$$F(x, \cdot g(f(\underset{\sim}{x})))$$

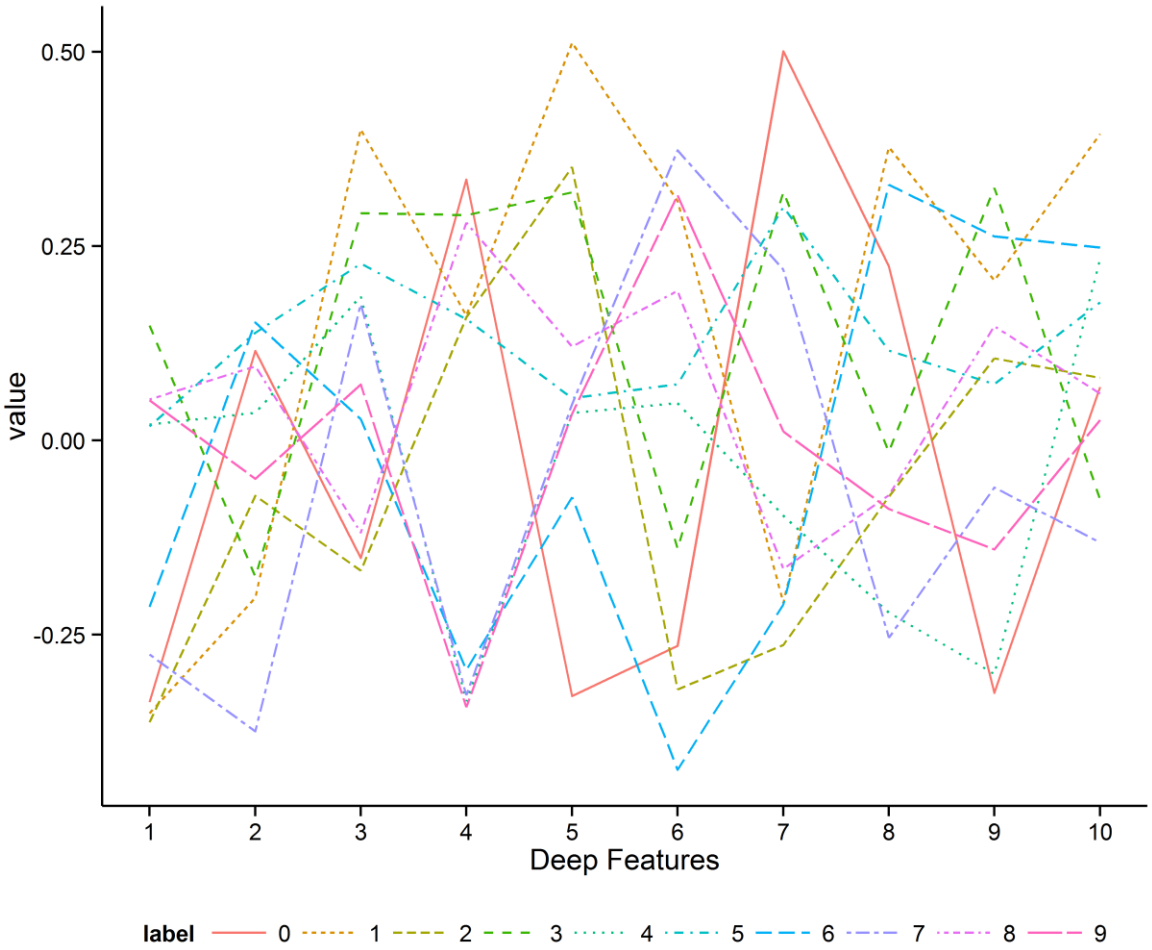
$$(\underset{\sim}{x})$$

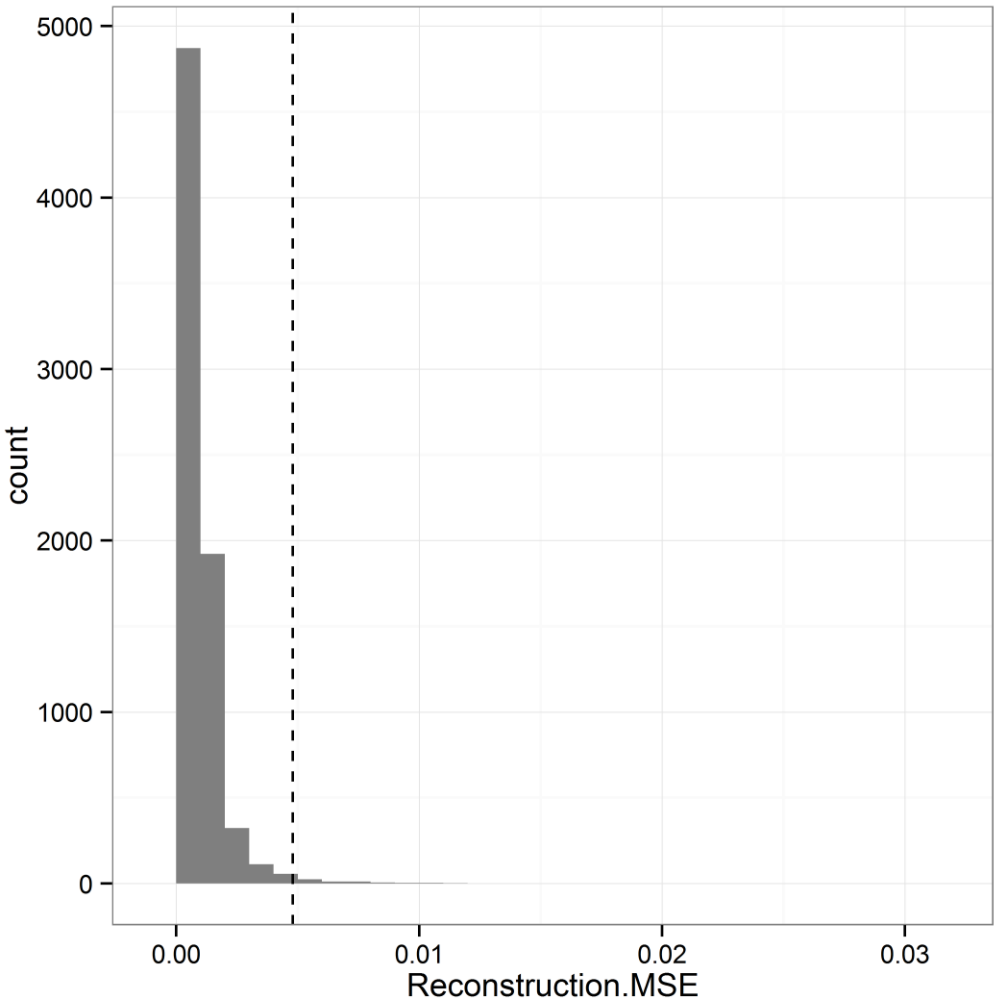


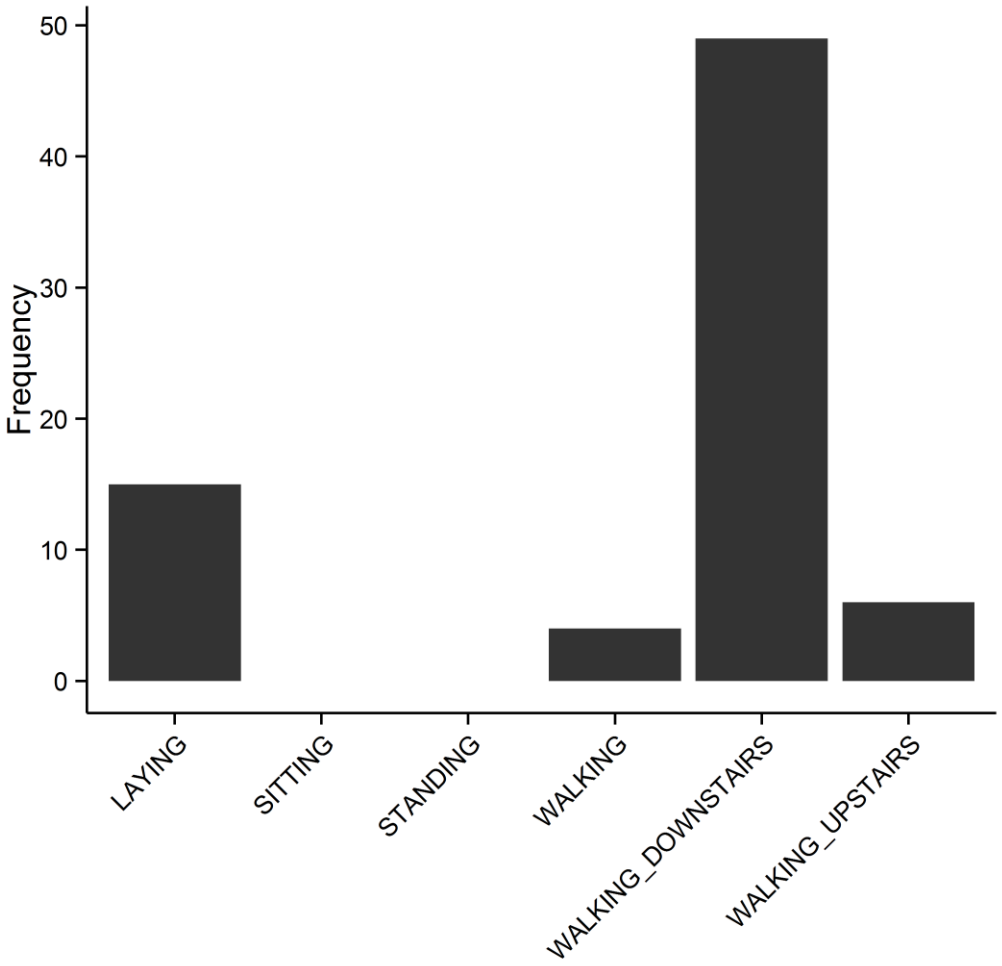

1
2a
2b
2c
count
3000
2000
1000
0
0.00
0.02
0.04
Reconstruction.MSE

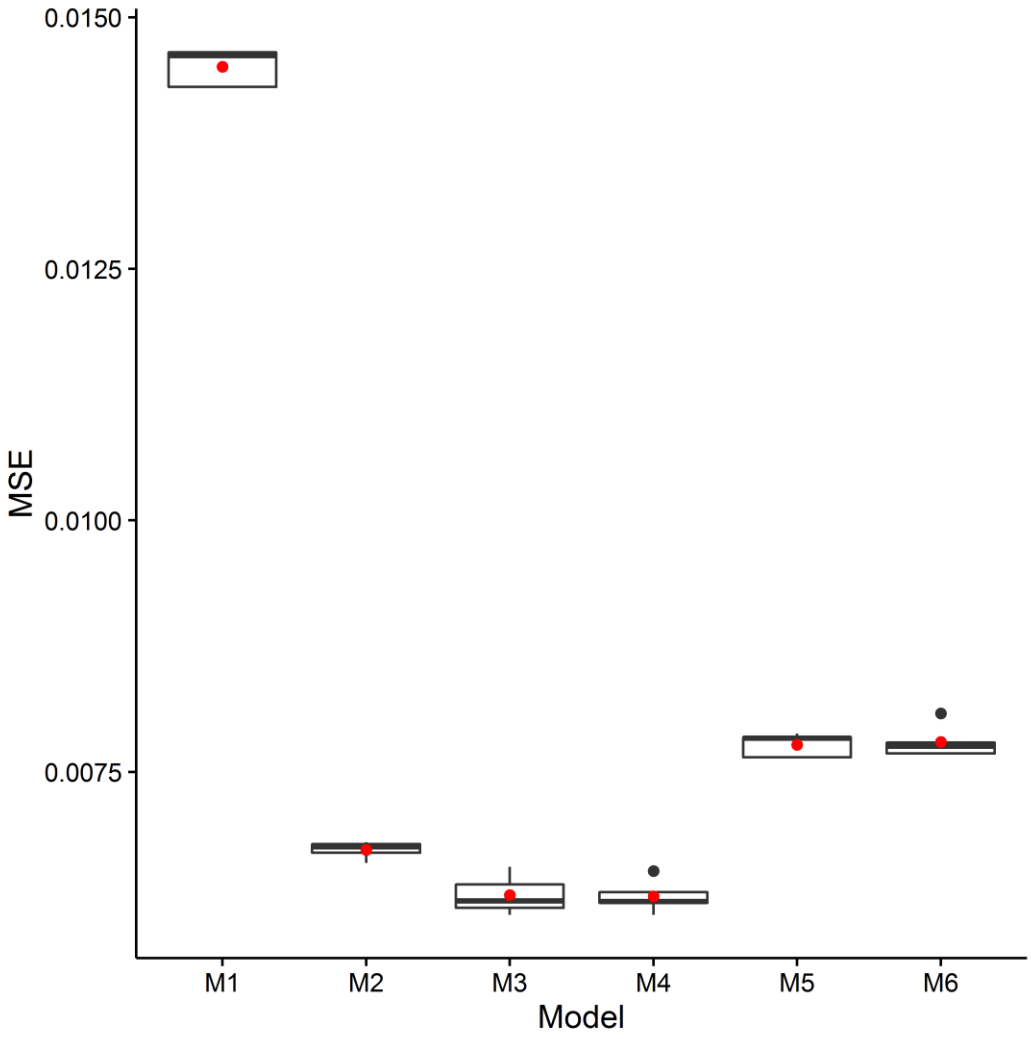

M1
M2a
M2b
M2c
0.005
0.020
0.035
0.01
0.03
0.05

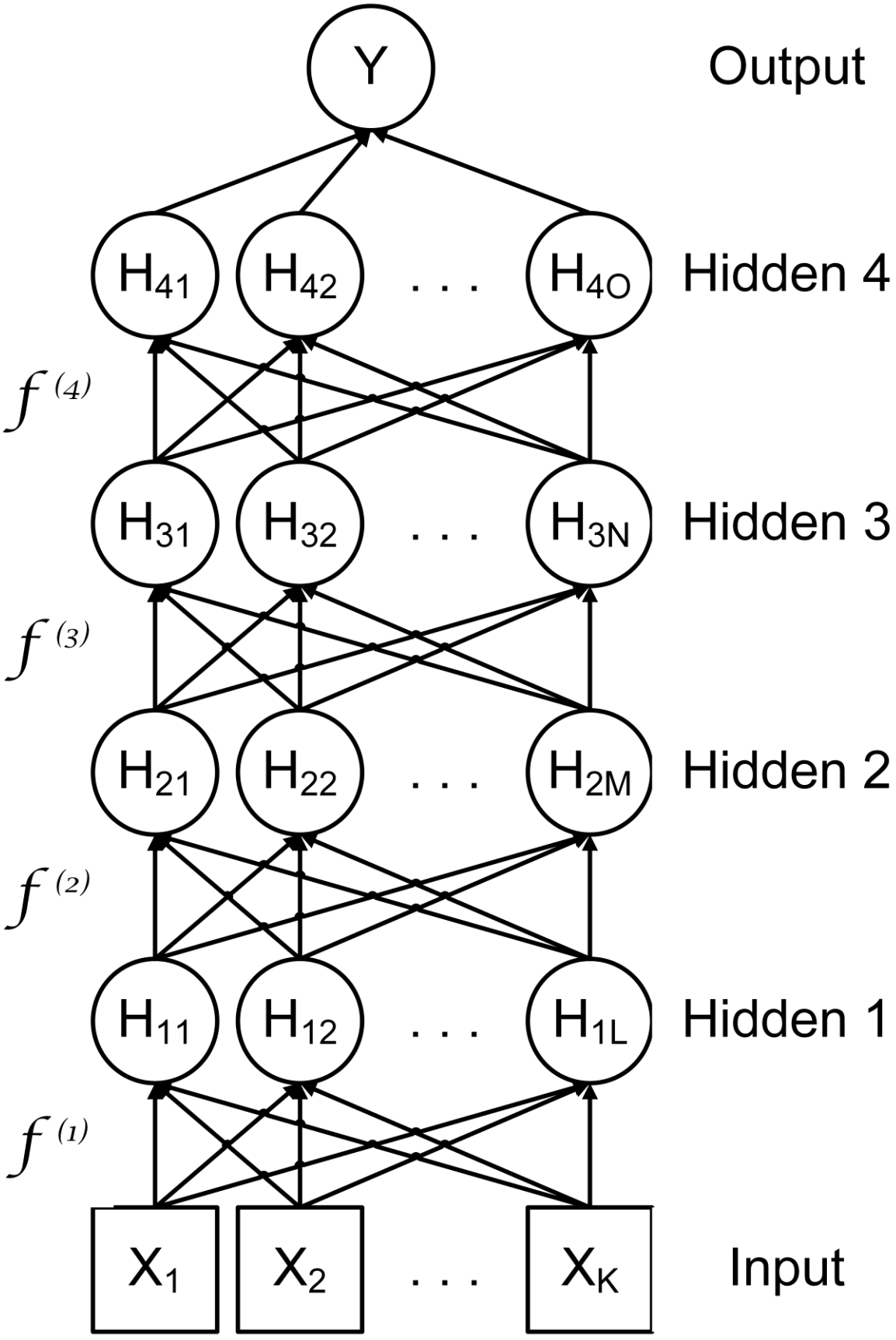

100
75
50
25
0
count
-0.50
-0.25
0.00
0.25
0.50
r

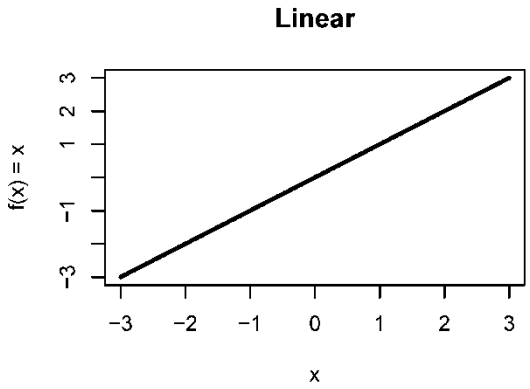

0.50
0.25
0.00
-0.25
value
1
2
3
4
5
6
7
8
9
10
Deep Features
label
0
1
2
3
4
5
6
7
8
9

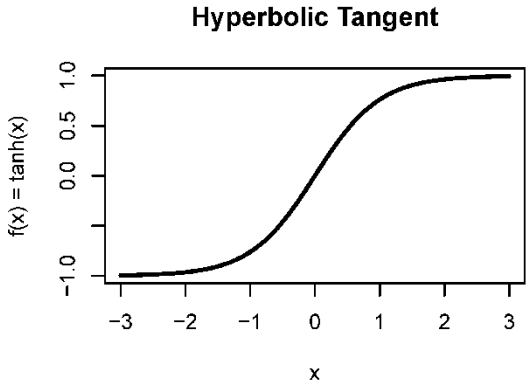

5000
4000
3000
2000
1000
0
count
0.00
0.01
0.02
0.03
Reconstruction.MSE

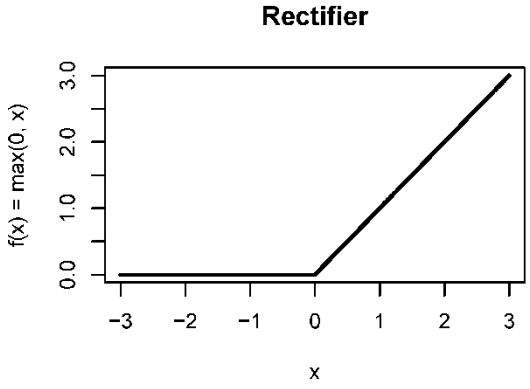

Frequency
50
40
30
20
10
0
LAYING
SITTING
STANDING
WALKING
WALKING_DOWNSTAIRS
WALKING_UPSTAIRS

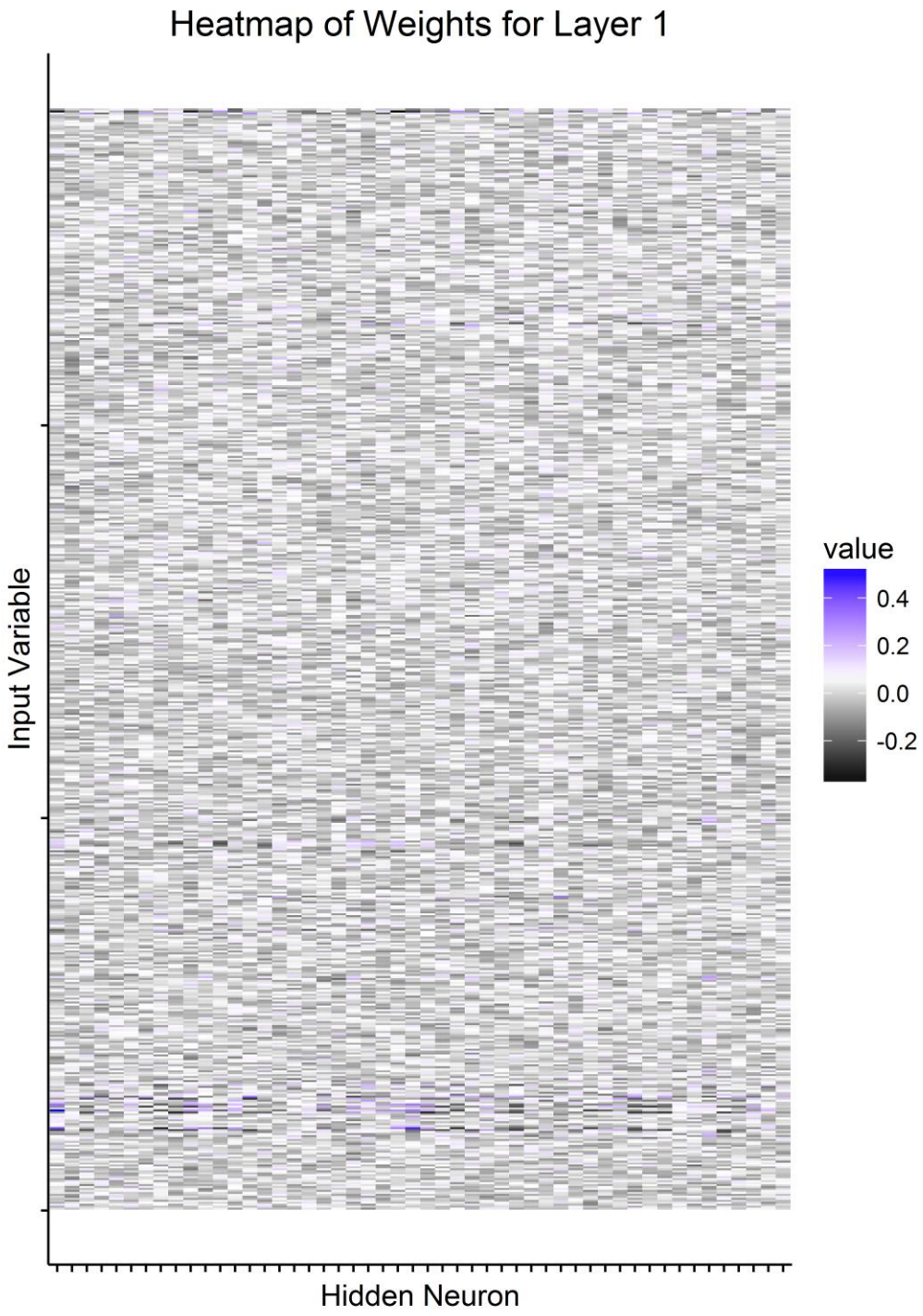

0.0150
0.0125
0.0100
0.0075
MSE
M1
M2
M3
M4
M5
M6
Model

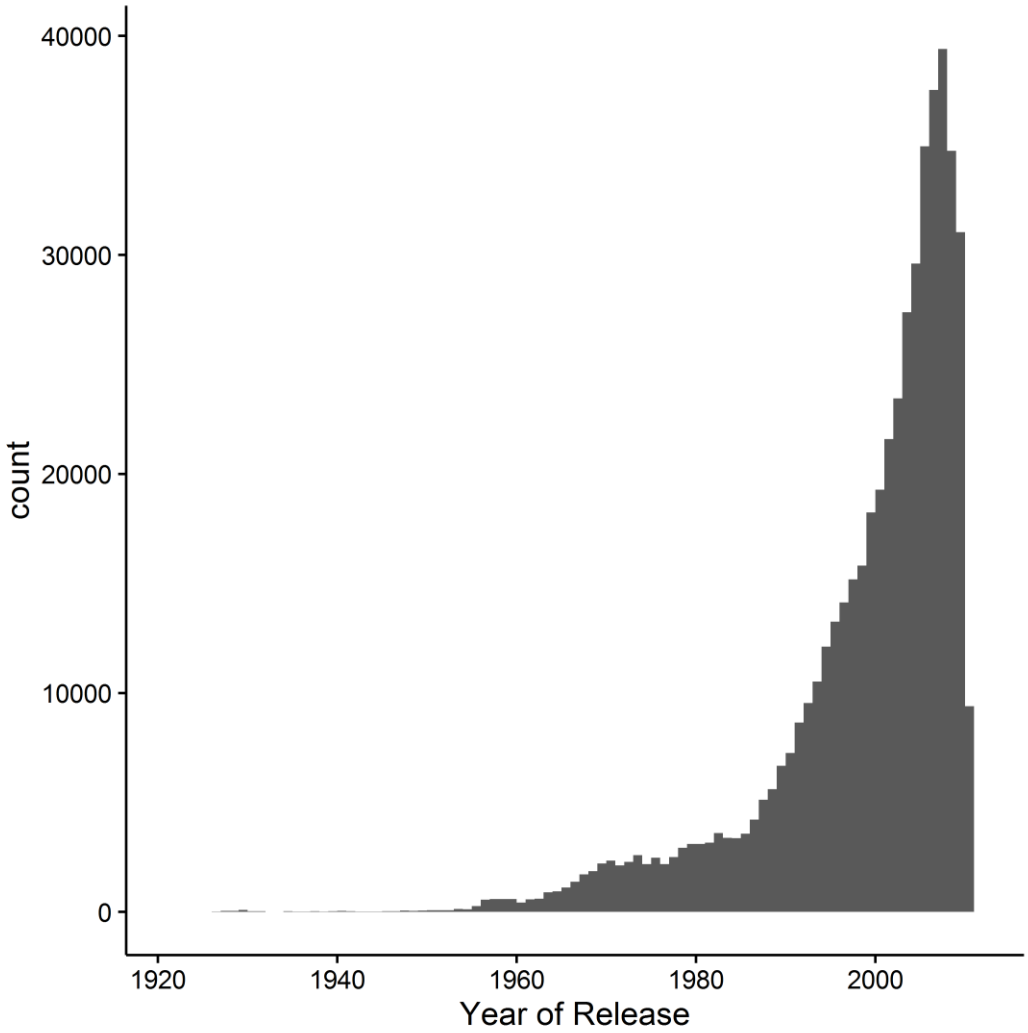

Y
Output
H41
H42
. . .
H4O
Hidden 4
f (4)
H31
H32
. . .
H3N
Hidden 3
f (3)
H21
H22
. . .
H2M
Hidden 2
f (2)
H11
H12
. . .
H1L
Hidden 1
f (1)
X1
X2
. . .
XK
Input

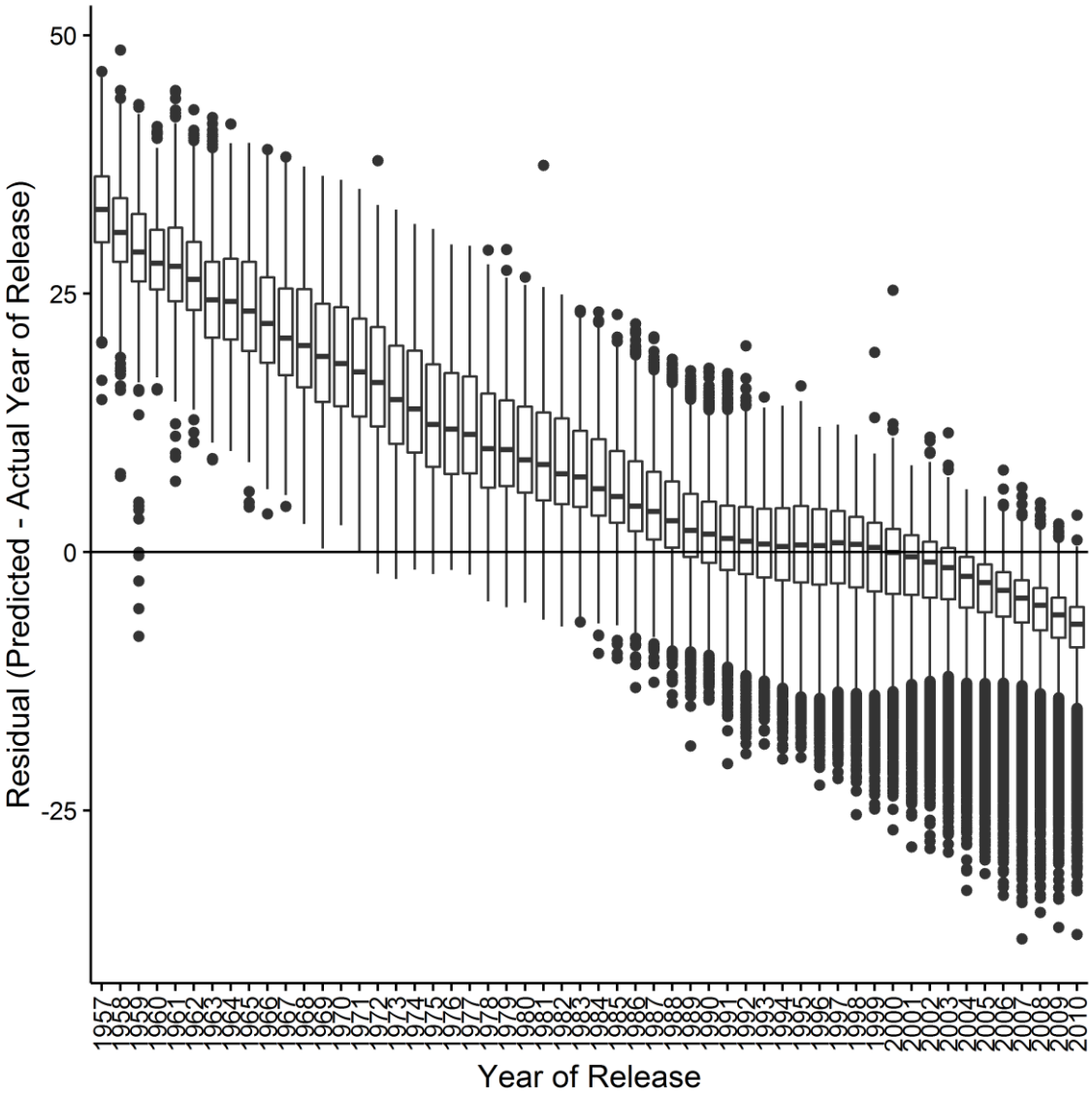

Linear
f(x) = x
−3
−1
1
2
3
−3
−2
−1
0
1
2
3
x

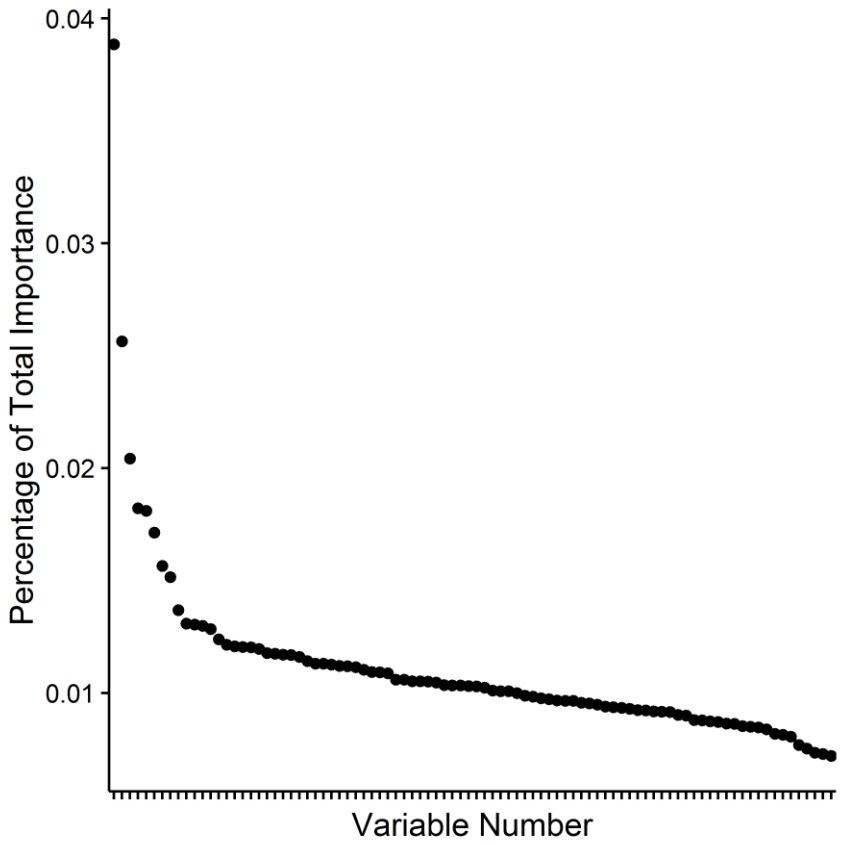

Hyperbolic Tangent
f(x) = tanh(x)
−1.0
0.0
0.5
1.0
−3
−2
−1
0
1
2
3
x


Rectifier
f(x) = max(0, x)
0.0
1.0
2.0
3.0
−3
−2
−1
0
1
2
3
x

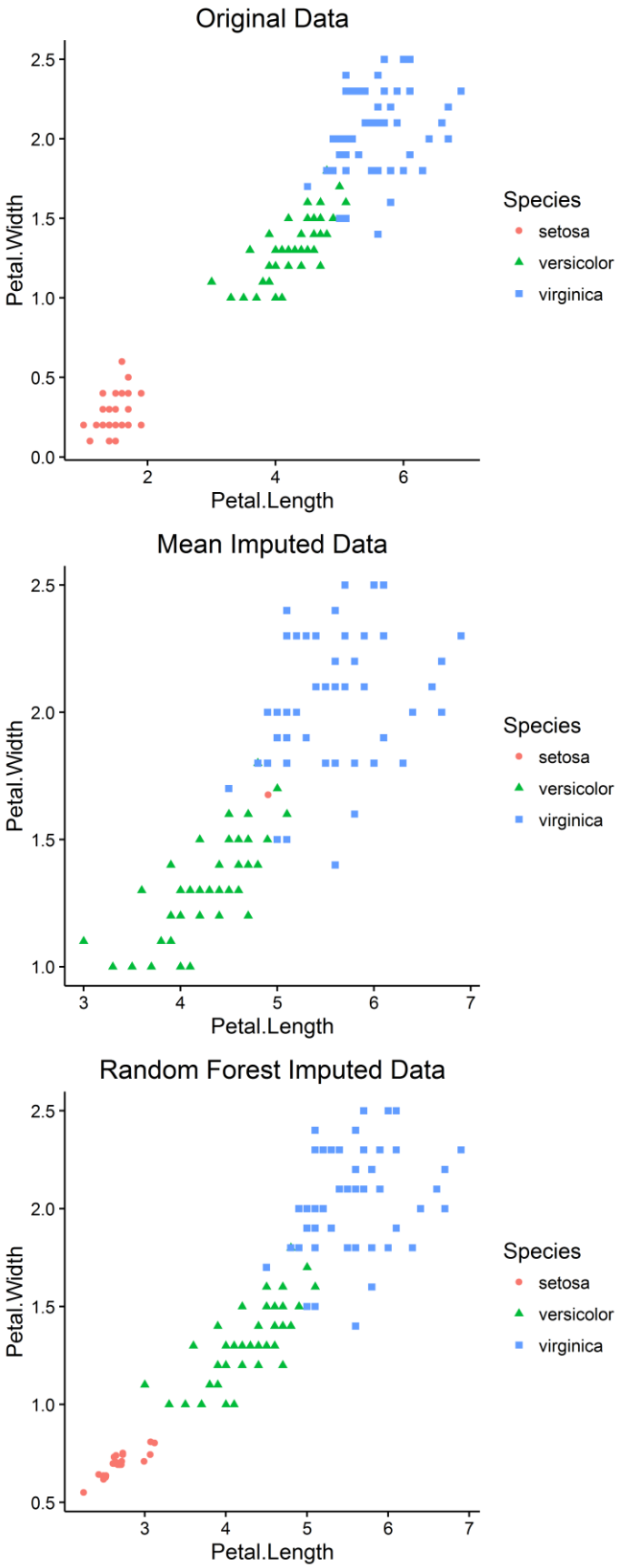

Heatmap of Weights for Layer 1
Input Variable
Hidden Neuron
value
0.4
0.2
0.0
-0.2

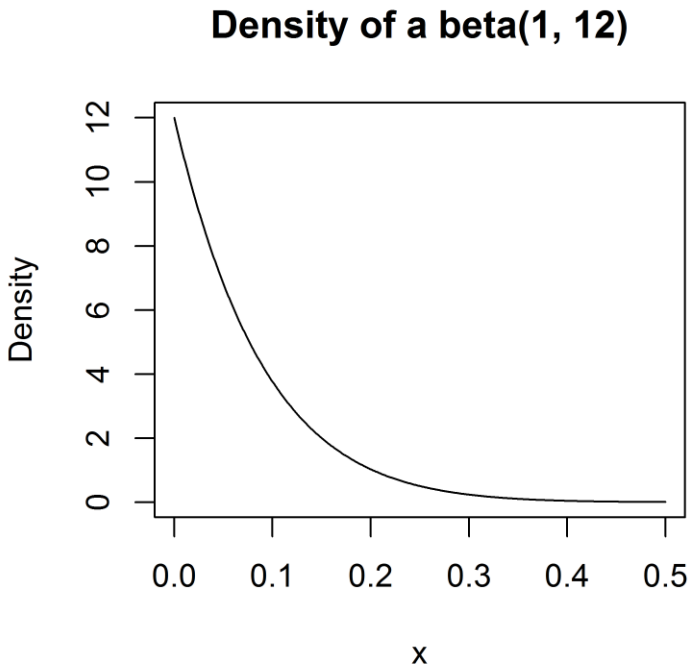

40000
30000
20000
10000
0
count
1920
1940
1960
1980
2000
Year of Release

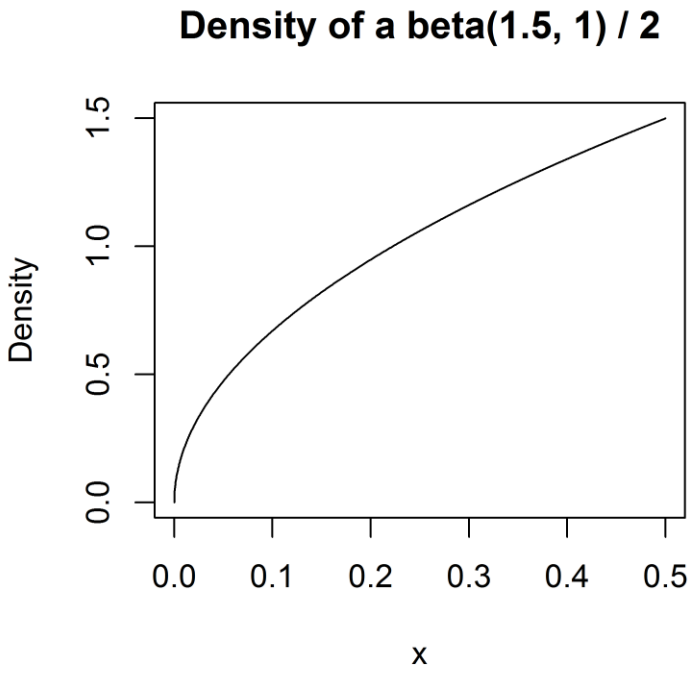

Residual (Predicted - Actual Year of Release)
50
25
0
-25
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
Year of Release

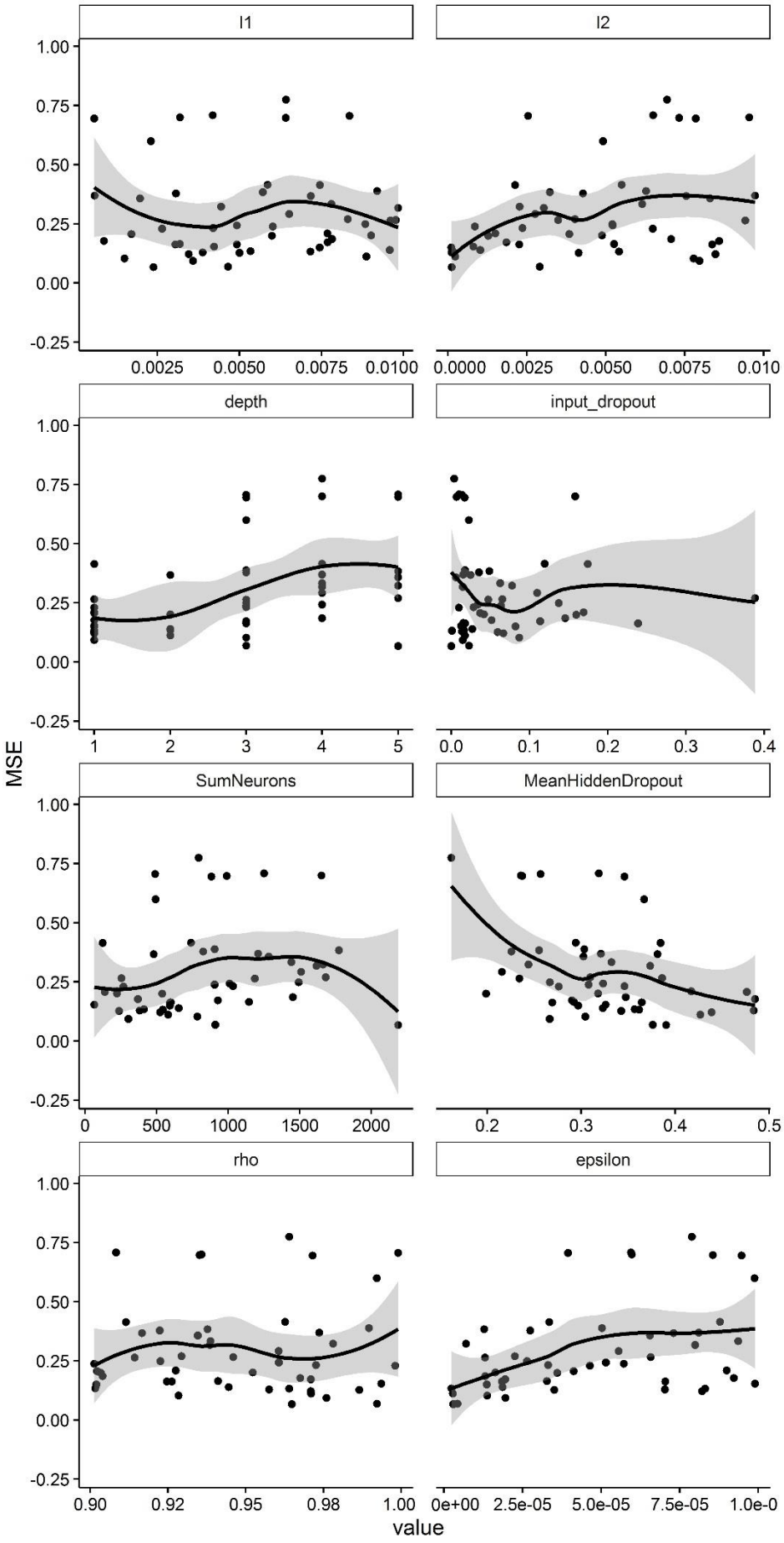

0.04
0.03
0.02
0.01
Percentage of Total Importance
Variable Number

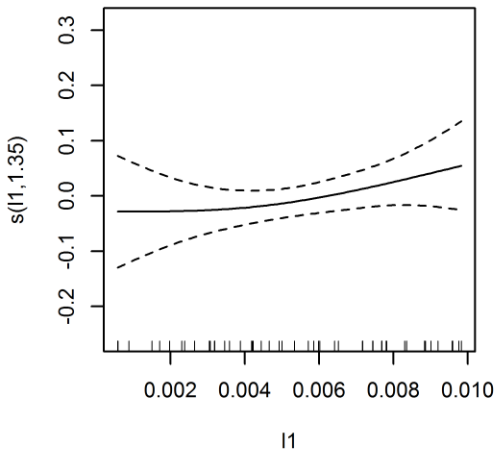
Chapter 6: Tuning and Optimizing Models

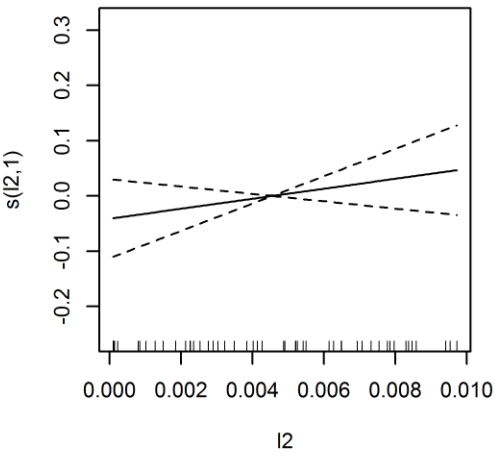


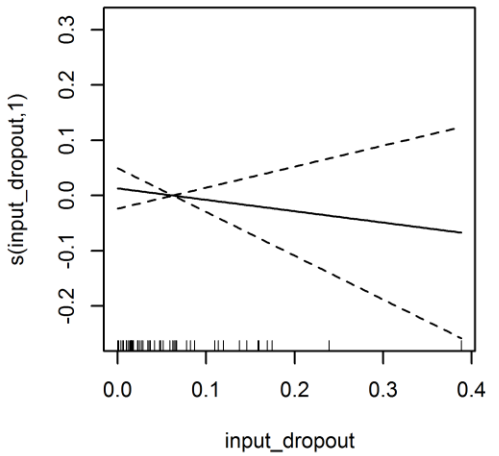

Density of a beta(1, 12)
Density
0
2
4
6
8
10
12
0.0
0.1
0.2
0.3
0.4
0.5
x

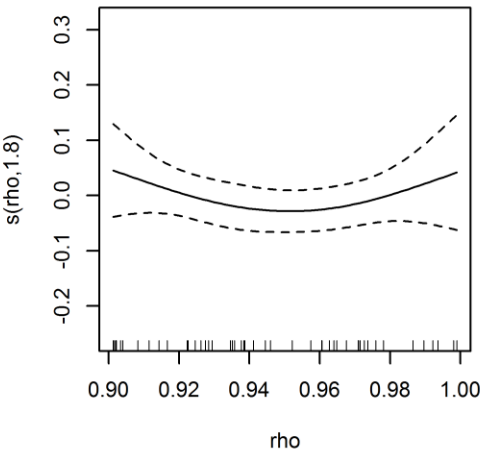

Density of a beta(1.5, 1) / 2
Density
0.0
0.5
1.0
1.5
0.0
0.1
0.2
0.3
0.4
0.5
x

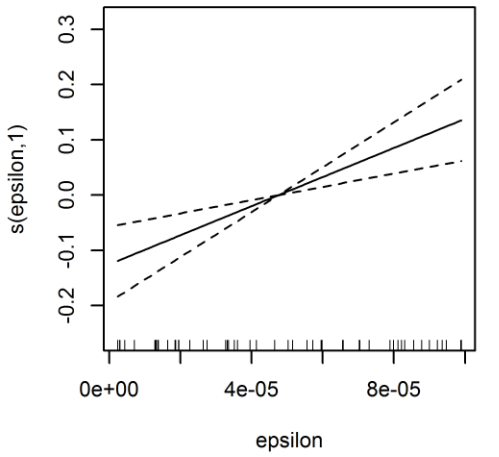

l1
l2
depth
input_dropout
SumNeurons
MeanHiddenDropout
rho
epsilon
MSE
value
1.00
0.75
0.50
0.25
0.00
-0.25
0.0025
0.0050
0.0075
0.0100
0.0000
0.010
1
2
3
4
5
0.0
0.1
0.2
0.3
0.4
0
500
1000
1500
2000
0.5
0.90
0.92
0.95
0.98
1.00
0e+00
2.5e-05
5.0e-05
7.5e-05
1.0e-0

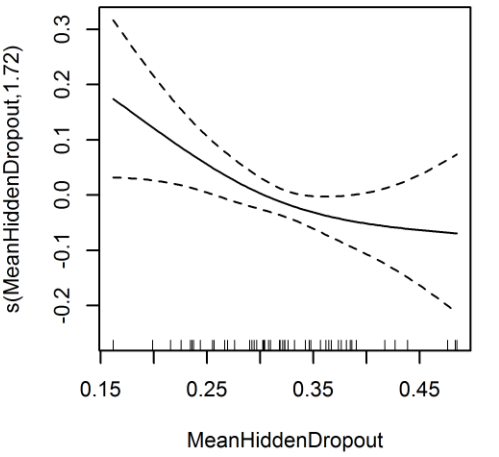

s(l1,1.35)
-0.2
-0.1
0.0
0.1
0.2
0.3
0.002
0.004
0.006
0.008
0.010
l1

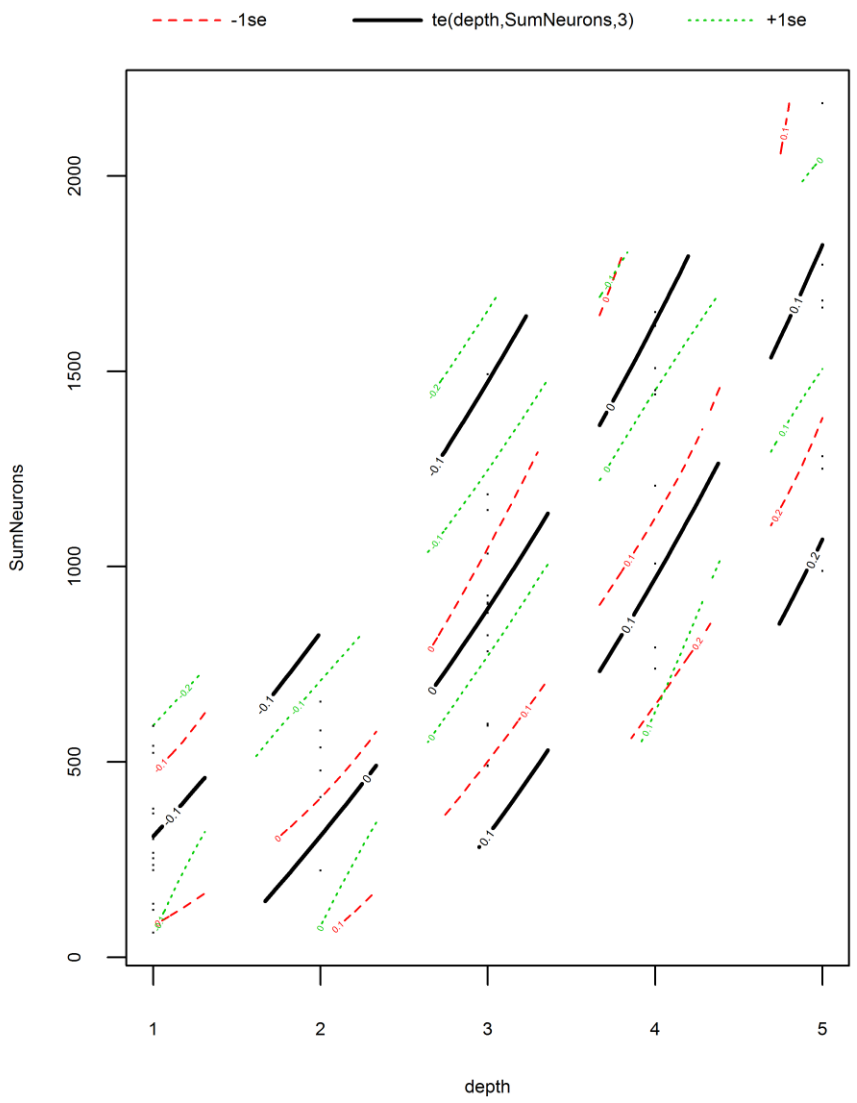

s(l2,1)
-0.2
-0.1
0.0
0.1
0.2
0.3
0.000
0.002
0.004
0.006
0.008
0.010
l2


s(input_dropout,1)
-0.2
-0.1
0.0
0.1
0.2
0.3
0.0
0.1
0.2
0.3
0.4
input_dropout


s(rho,1.8)
-0.2
-0.1
0.0
0.1
0.2
0.3
0.90
0.92
0.94
0.96
0.98
1.00
rho


s(epsilon,1)
-0.2
-0.1
0.0
0.1
0.2
0.3
0e+00
4e-05
8e-05
epsilon


s(MeanHiddenDropout,1.72)
-0.2
-0.1
0.0
0.1
0.2
0.3
0.15
0.25
0.35
0.45
MeanHiddenDropout


-1se
te(depth,SumNeurons,3)
+1se
SumNeurons
0
500
1000
1500
2000
depth
1
2
3
4
5