A picture containing text, clock

Description automatically generatedText, table

Description automatically generatedA picture containing text, clock, watch

Description automatically generated

If linearity assumptions are met

In regression we have: (which allows for writing R^2 differently)

LHS is logit transform of π

(Linear) logistic regression

Linearity assumptions (for LS regression)

Text, letter

Description automatically generatedDiagram

Description automatically generated with low confidence

The following removes col. upo3:

For J>2

A picture containing text, screenshot

Description automatically generated

adds log(03) as a col. and removes 03

Likelihood (for Bernoulli x then for Binomial)



If in addition ε1,.,,εn are uncorrelated and ~ N(0, σ2) or large n

(In-sample) R2 in R

Text, letter

Description automatically generatedText

Description automatically generated



LOOCV in R





Scatterplot

Text, letter

Description automatically generated

Nadaraya-Watson (normal kernel)

Non-parametric regression in R

(Cross-validated LOOCV) R2 in R

Multinomial logistic regression

will give the probability of heart disease (N/m)

A picture containing text, orange, dark

Description automatically generated

Does indicator variable affect thorax length?

Reject H0 if Pr(>F) <= 0.05

Text

Description automatically generated

where the probabilities of each class are stored in the columns (columns same order as levels (…)) but we can also achieve logistic regression by using dummy encoding and glm:

Local polynomial

Text, letter

Description automatically generated

Order data

Text

Description automatically generated

MARS in R

Smoothing splines



A picture containing logo

Description automatically generated

Q-Q plot



ROC and Misclassification graph in R

Text

Description automatically generated

We can then find their R2 as follows:

TODO: Hat matrix (serie 5 methods and non-parametric regression)

A screenshot of a computer

Description automatically generated with low confidence

TODO: Equivalent ways of calculating LOOCV using hat matrix and manual computation

Bootstrap in R



which plots the TPR vs FPR. To plot the misclassification rate as a function of threshold prob. (i.e., after this it is considered positive)

Text

Description automatically generated with medium confidence

MARS fits a function of the form





K-Fold CV in R

Text

Description automatically generated

Mallows Cp statistic

The MSE can be estimated by:

Text, letter

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generated with medium confidenceText

Description automatically generated with medium confidenceText, letter

Description automatically generatedA picture containing text, watch

Description automatically generatedA close-up of a clock

Description automatically generated with low confidenceText

Description automatically generatedChart, box and whisker chart

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generatedText

Description automatically generated with medium confidenceA picture containing text, clock, watch

Description automatically generatedA picture containing text, clock, watch

Description automatically generatedA picture containing text, watch

Description automatically generatedA picture containing text, watch, gauge

Description automatically generated

Bootstrapping LDA/QDA (both part of MASS library)

which is prop. to Cp(M):

Cp in R (note: object is of type fit, and sigma can be found using sigma(fit))

Forward selection (in case p, the nr. of predictors vars. Is too large for exhaustive search):

1) start with smallest model

2) add predictor which reduces the MSE the most

3) repeat 2 until all predictors or large nr. of predictors selected (now a seq. models is produced)

4) choose the model in the seq. which has smallest Cp statistic.

Backward selection is similar.

Note that the linear regression model does not assume a normal distribution for the predictors, but a skewed distribution and outliers often result in  
regression solutions that are largely determined by very few points.

Forward and backward selection in R

Tukey Anscombe is residuals versus fitted values. Q-Q is empirical quantiles vs standard normal quantiles.

Whereas Q-Q plot is given by

For lm, Tukey Anscombe is given by



Maybe add all-subsets regression in R (from HW2)?

Add anova table and f-test and interpretation

Poly. regression using a polynomial of degree d (no interaction term):

Neural Networks (1 Layer with q units)

R2 describes the proportion of the total variation of the response Y around its mean Y bar which is explained by the regression Y hat.

To get the weights use summary(Nfit). R prefixes input variables with *i* and hidden unit with *h*.To get the error use sum(residuals(Nfit)2)

Make sure to scale the data (except for response) to avoid getting stuck in the flat regions of the sigmoid function. Weight decay is good as it reduces the dependency on the starting values and reduces the importance of choosing the # of hidden units

and f0 is typically identity for regression and sigmoid for classification. We can add a linear regression component (skip conxn):

Alternatively,

where

We can also get the 3D plots (effect of the interaction of 2 terms on the response) using

summary (mars fit) gives the importance of each variable. If we fitted a model of degree 2, then we can get the graphs of the response variables as a function of each predictor variable as follows:

To plot the same thing but for lm use:

To plot the splines do:

Generalized additive model in R

And then pred.cv <- prediction (all.y.pred, all.y.true) and proceed as normal (but add avg = “threshold” for ROC and avg = “vertical” for misclassification) use add = T to add them to same plot. One can specify the cost for FN and FP using cost.fp = 0.8, cost.fn = 1.2 as args to performance().

To plot the average ROC (across all folds) initialize a vector of lists and assign them as follows:



Decision boundary plot in R



Decision boundary between class 0 and 1:

QDA Classifier (more easily overfits if p is big)

if the covariance matrix ∑ is diagonal and const. then w is parallel to the line connecting µ0 and µ1



Prediction of LDA found by taking argmax: of:

LDA Classifier

Bayes Risk Bayes Classifier

Expected size of out-of-bootstrap sample [roughly 1/3 of points will be out of sample]:

“basic” = “Reversed quantile”

“norm” = “Normal”

“perc” = “Quantile”

A picture containing clock, watch

Description automatically generated

Z value = (µ - µ0)/SE

(Square of) residual standard error

where µ0 is the mean under the null hypothesis (0 for coefficient estimates) and µ is the measured mean

Note that the ridge estimate for the coefficient vector beta will be biased, but has less variance than the LS estimate (obtained when λ=0)

is equivalent to using a neural network (note results may differ due to different seed). Optimization in R

A picture containing text, clock, watch

Description automatically generated

Assume we have a fct. like this for the neg. log likelihood which we want to minimize (case of binomial distribution of response and logist. regr.),

derived from SV Decomposition of X as UDVT where U is n\*n, V is p\*p, and both are orthogonal. D is n\*p diag.

Text, letter

Description automatically generated

In case of violations of item 3, we can use weighted least squares instead of least squares. Similarly, if item 4 is violated, we can use generalized least squares. If the normality assumption in 5 does not hold, we can use robust methods instead of least squares. If assumption 2 fails to be true, we need corrections known from “errors in variables” methods. If the crucial assumption in 1 fails, we need other models than the linear model.

If X has full rank p, then *X* is p-dimensional subspace:

Linearity assumptions violations

Unlike ridge regression, solved by differentiating matrices, lasso regression requires quadratic programming. Has the goal of variable selection.

Lasso Regression

**Several extensions** are elastic net regression:

Then the following will return the vector beta (same as that of logistic regression),

Text, letter

Description automatically generated

which is equivalent to solving the problem:

Trees (CART)



where,

and the constraint is called **elastic net penalty**



which lies on an n-p dimensional subspace (complement space of *X*)

For all α in [0,1), the elastic net penalty is singular (no first derivative) at 0 and it is strictly convex for all α > 0, note that the lasso penalty (α=0) is convex, but not strictly.



**Another extension** is the **adaptive lasso** (shown to have oracle properties)

A picture containing object, clock, watch

Description automatically generated

Multiple regression vs single predictor



Suppose β hat is sqrt(n) consistent estimator of β (for example the LS estimate), pick a γ>0 and define wj hat as 1/|βj hat| then adaptive lasso is given by

A picture containing clock, watch

Description automatically generatedGraphical user interface, application, Word

Description automatically generated

ANOVA can be used to see if ANY variable is signif.

The cost function is: (note that the size of the tree is the # of leaves, n, and that for a binary tree # of nodes = 2n - 1 ). For regression, R(T), can be -2\*(log likelihood)=sum of squares as in the case for regression or the misclassification rate for classification

which is a strictly convex optimization problem (i.e. has no local minima) and can be solved efficiently

Text

Description automatically generated

ANOVA relies on the F-statistic:

The final extension is the **relaxed lasso regression:**

A picture containing text, watch

Description automatically generated

**Pros** of trees: Interpretable, can deal with missing values using “surrogate” split, does automatic variable selection.

**Cons**: piecewise constant (for probability estimate or regression). Unstable split: if a split is “wrong”, then all of the splits below it will be “wrong”.

Text

Description automatically generated

Lasso and ridge in R



Is a confidence interval which covers the true beta with probability 1-alpha

Combine two formulas and generate all deg <=3 interactions in R

Find Numerical Columns in R

CART in R

Text

Description automatically generatedA picture containing shape

Description automatically generated

Generalized least squares (errors correlated with covariance matrix known up to scalar factor):



NW Kernel Estimator

cp stands for cost complexity pruning (i..e penalizes large trees -> higher cp means shorter trees. cp = alpha/R(0)).

Company name

Description automatically generated with low confidence

Plotting can be done as follows (note that each node will display J numbers for each category which falls into this region in the order given by levels(data$var)):

A picture containing text, watch, clock, gauge

Description automatically generated

Generate model frame [data frame containing all terms required by a formula] and design matrix:

Graphical user interface, text

Description automatically generated

To get NW with optimal local bandwidth selection use library lokern, function lokerns.

**One-standard-error rule**: Find the model with the lowest cross-validation error. Then choose the simplest model, which is at most one standard-deviation worse than that model.

To prune the tree according to this rule, we need cross validation error of the tree, use (ignore relative error column):

Text

Description automatically generated

The covariance of the estimator in terms of smoother (hat) matrix:

**Ridge and lasso fitting** (and plotting coefficients vs log lambda)

Text, letter

Description automatically generated

A picture containing text, clock, watch

Description automatically generatedText, letter

Description automatically generated

For non-parametric regression, degrees of freedom = trace of smoother S matrix

Bias variance decomposition

Text, letter

Description automatically generated

CV for Ridge/Lasso and Saving Objects note that the 1-std error rule was used to get λ (the λ that minimizes CV is given by cv.eln$lambda.min)

Text, letter

Description automatically generatedText

Description automatically generatedA picture containing text

Description automatically generated

Local polynomial non-parametric regr.

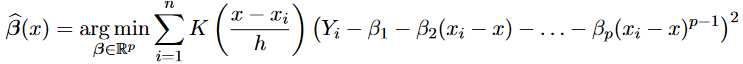
Bootstrapping trees:

Ridge regression is formalized as:

Ridge Regression

which is equivalent to solving [via Lagrange mult.]:

can be rewritten as





A picture containing text, watch

Description automatically generatedA picture containing text, watch

Description automatically generated

Smoothing spline (3(n-2)+2 constraints, 4\*(n-1) coeff)

which is beneficial for ridge since we do not want to penalize the intercept term (β0’ is simply Y bar, the avg of the Y values).



Formulas:

Note that XtX + λI is always inv. for λ > 0

Set precision in R

Diagram

Description automatically generatedText

Description automatically generated

A screenshot of a computer

Description automatically generated with low confidenceText

Description automatically generated

Legend in R