Sum up

Linear models

Linear model aims

Linear model are used to describe reality, to explain phenomena, to predict the behavior of variables.

Assumptions

- Linear relation between X and Y
- Independence of the residuals (no incr/decr trends in residual plots)
- Homoscedasticity the residuals have the same level at each X
- Normality on the residual of X and, by reflection on Y

What is P-value

P-value is an indicator of the statistical significance of an hypothesis against another one in a statistical test. The p-value is a probability and ranges from 0 to 1. It is the probability, under the null hyp, of observing some more extreme events in the direction(s) of the alternative hypothesis. Therefore, when P is very close to 0, we have strong evidence to reject the null hypothesis. When it is near to 1, we have strong evidence in failing to reject the null hypothesis. The common significance thresholds are .01, .05, .1. The p value itself is not sufficient for conclusions. There is a chance that the sample chosen is not representing the entire population, maybe for a sampling error or a poor sample.

F test

It's a statistical test and it's used to compare a model with no predictors (only intercept beta 0) with a model with different predictors. Null hypothesis: model without covariates is the same as model with covariates. How large does the F -statistic need to be before we can reject H 0 and conclude that there is a relationship? It turns out that the answer depends on the values of n and p. When n is large, an F -statistic that is just a little larger than 1 might still provide evidence against H 0 . In contrast, a larger F -statistic is needed to reject H 0 if n is small. When H 0 is true and the errors ε i have a normal distribution, the F -statistic follows an F -distribution. if we use the individual t-statistics and associated p-values in order to decide whether or not there is any association between the variables and the response, there is a very high chance that we will incorrectly conclude that there is a relationship. However, the F -statistic does not suffer from this problem because it adjusts for the number of predictors.

It's more important than R squared because if the model is not statistically significant, R square is useless. If the model is significant but R square is low, the model is not so good but it is however useful.

Residual plot analysis

To err is human, to err randomly is (statistically) divine. Residual analysis is an important procedure to validate the goodness of a linear model. The residual analysis could reveal a pattern in the residual charts, and this could mean several things:

- we're ignoring a covariate that needs to be introduced in the model
- we're ignoring a hidden relation with a covariate we ignore
- data are not homoscedastic
- data are not in a linear relation, maybe they need a transformation
- we're missing an interaction of variables inside the model

Residuals must not have a pattern themselves (autocorrelation) and should be normally distributed Residuals must not be related in some ways to a variable.

RSS

The method of determination of the coefficients of a linear model is minimizing the residual sum of squares. A residual is the diff from the observed y and the predicted y hat. The sum of squares indicates a positive quantity that gives us the idea of how the model fits the reality. Conceptually, the RSS will never be 0. RSS is not useful itself, but can give us the idea that a model fits better than another if RSS is lower.

R squared

Coefficient of determination. It's a ratio (0 to 1) and gives us the idea of how the model is explaining reality. It's the ratio between the explained deviance divided by the total deviance. R^2 is 74.3%. The variation in Y is explained by 74.3% by using X. R squared is very important for prediction, not so much for explaining relations. Better to use adjusted version, because too many predictors could introduce noise. When performing a linear regression on many variables, whenever we add a variable to the model the R^2 increases, even if the variable does not better the model.

Standard Error

Average distance of the observed values from the model line. It goes from 0 (never) to + infinity and it's expressed in the units of Y. When we predict something we're more interested in SE rather than R^2 because we understand how big are the mistake we can make in terms of units.

Multi-Collinearity

It is a phenomenon that happens when 2 or more predictors are not completely independent, so they are in a relation of some kind. The main effect is that the extended model has some additional noise inside, the coefficient can differ from the base model. Sometimes the R^2 is even higher because the model fits better, but the statistical significance of the covariates is lower. The most common and easy solution is to try to remove one or more covariates that are multicollinear. In this way, the model could fit better the reality. Even if, in some prediction models, small multicollinear predictors are left.

Anova

Anova is a function that performs the analysis of variances. It can be used to compare two different nested models together. In this case, we perform an F statistical test. The null hypothesis is that the two model don't differ each others, and therefore, the new covariates are not significantly useful. In case P-value is very high, we fail to reject the null hypothesis. When P-value is low, we can reject the null.

Error Types

The aim of a statistical test is trying to reject a null hypothesis or the alternative hypothesis. There can happen different kind of error because we're dealing with a sample of data and hence, with uncertainty. First type error is to reject the null hypothesis when it is not the case. Second type error is to fail to reject the null hypothesis when we should reject it.

LDA

Linear Discriminant Analysis is a technique of dimensional reduction (thus similar to PCA). It creates a linear combination of variables wrt known categories. These new combinations are called Linear Discriminant LD1, LD2... For example, imagine we have 2 dimensions and we see some blue/red points. We could create a line/axis that maximize the blue/red points separation. In practical terms, we maximize the mean separations and minimize the blue/red scatter. This new line is the LD1 and it's useful to reduce dimensions. It's also useful for categorical predictions: given X and Y, is this point blue or red?

PCA

PCA is a technique of dimensional reduction. We aim to identify a linear combination of variables that captures most of the variability of the variables. It is PC1. A linear combination is a cocktail of covariates in different ratio (loading scores). E.g. X,Y,Z. PC1: 0.97X + 0.2Y + 0.3Z. It means that 97% of X explains PC1 separation

Through PCA we can plot 1000 dimensions onto 2 axes. With a biplot, we can also overlap info about covariates and which covariate is responsible for that PC axis.

Model selection

- RSS - of course the lowest the better, but many times, more complex model have lowest RSS because they tend to use noise to overfit
- Anova if models are nested
- Adjusted R^2
- CV - K-fold or LOOCV (number of folding). We can calculate the error of our model estimating a different dataset of data (based on CV). And we get an error. We compare 2 models errors. If delta is positive E2-E1, then it's better model M1 and viceversa.
- AIC - Akaike information criterion, it's a way to estimate the goodness of a model from the # of independent variable and how well the model describes data (max likelihood). More Complex Models are penalized. The lowest AIC, the better. R^2 can be similar for 2 models, use AIC!
- BIC - Bayesian I.C. like AIC, the lower, the better. AIC or BIC? Most of the time they agree. Use both to support evidence and in case they differ, report it.

Ridge and lasso

We build a linear model. If the training set is small, we risk over-fitting it using least square method. e.g. with 2 points! We can solve the problem, by minimizing the ResidualSS + lambda * slope^2. Where lambda is a penalty. In this case, we're doing a ridge regression. lambda = 0 => un-penalized normal linear model, lambda = inf => max penalty, model has a slope=0. How to choose lambda? using CV! Lasso is similar to ridge, always same penalty mechanism, but the diff is that we drop useless covariates by setting their coef to zero. Useful in model selection, where we have many covariates. But ridge give of course better results.

Steps of analysis

1. EDA
   A. Read carefully the scientific question we want to answer and read the data set variables descriptions
   B. Is the response variable a factor? Is it a continuous variable?
   C. Is the response variable normally distributed? If not, and the observations are few, it's OK, justify it saying this. If not, apply a log transformation. Plot hist of before and after to justify that transformation.
   D. Are there any N/A values? Remove those records
   E. Are there any factors which are not factor? Convert them.

- F. Does the scientific question ask us to shrink the data set? Create a new one from the old one.
- G. Plot response variable wrt factor variables using box plots. Plot interactions between factor variables (if more than 1)
- H. Use pairs to investigate correlations between variables or plot scatter chart between response variable and continuous variables (by hand)
- I. Write down if you notice any linear/quadratic/non-linear trend

2. Find a model with a subset of variables for a continuous response variable
   - A. Use standard LM
     - a. Create a model with all the possible interactions
     - b. Delete the least significant from the lm
     - c. If in doubt between 2 nested models, try to use ANOVA
     - d. Try to add polynomials
     - e. Try to add natural splines on least significative variables (clouds of points)
   - B. Extend the LM using gam
     - a. Set a seed and write it in the report
     - b. Install and import gam
     - c. Get DoF using cross validation of a smooth spline
     - d. create a GAM with that smooth spline with rounded DoF
     - e. Do it for the different variables
   - C. Validate the model
     - a. Perform residual analysis
       - i. Plot the residuals
       - ii. Are the residuals equally distributed?
       - iii. Are there any leverage points? Possible outliers?
         1. In case of outlier, remove them
         2. Re-calculate models with new dataset
     - b. Plot the predicted vs observed values with red bisector line
     - c. Comment the result
     - d. Answer the scientific question
   - D. Discriminate between models
     - a. Are you still in doubt between models?
     - b. Use `anova` if nested models
     - c. Use deviance(model) to calculate the explained deviance. Higher is better
     - d. Use extractAIC(model) to calculate the Akaike Inf Criterion. Lower is better
     - e. Compare graphically the two observed vs predicted plots
     - f. Choose the simpler one
3. Find a model with many variables for a continuous response variable
   - A. Perform PCA - Principal Component Analysis
     - a. Apply func `prcomp`
     - b. Plot PCs using `biplot`
     - c. Do you see any horizontal, vertical vectors?

d. Are point separated into clouds? Maybe adding colors according to classes?

e. How much deviance is explained by the first N PC? Plot it

B. Perform PCR - Principal Component Regression

a. Set seed

b. Apply `pcr` func

c. Check deviance explained by the first N PCs

d. Plot the predicted vs observed

C. Use regularization

a. Set seed

b. Identify X and y (project on dataset columns)

c. Import glmnet lib

d. Ridge (best for predictions, no variable selection)

i. Apply `glmnet` func with alpha=0

ii. Apply `cv.glmnet` func with alpha=0 using lambda.min

iii. Apply `cv.glmnet` func with alpha=0 using best.lambda

e. Lasso (slightly worse for predictions, good for variable selection)

i. Same as ridge with alpha=1

ii. Check `coef` to analyze the variables selected

f. Plot error curves for cv

4. Find a model for a binary or discrete response variable (classification)

A. Analysis

a. Is the response variable binary/dycotomic or discrete?

b. Are the covariates few or many?

B. Modelling binary classification (yes vs no)

a. Try to use the glm using family=binomial

b. Try to add interactions and polynomials as you'd do with linear model

c. Compare models with `anova` (`chisq` test)

d. If in doubt use the simpler one

e. If some variables are not normally distributed or see clouds of points, use GAM with splines

f. Apply `gam` function passing `s(Variable, Degree)`

g. If in doubt between models check `deviance(model)` or `extractAIC(model)`

C. Validate binary classification

a. Assumption: Only 2 Classes (yes/no)

b. Split training/test data set

c. Apply model to the test set

d. Calculate yes/no vs predicted yes/no with a misclassification table (confusion matrix)

e. Install package and lib pROC

f. Plot ROC curve and discuss it

g. Check and discuss sensitivity and specificity

D. Modelling with discriminant analysis

   a. Assumption: Classes are well-separated (no overlapping)
   b. Install MASS package
   c. Apply `LDA` func
   d. Plot LDA model and analyze it
      i. Are classes vertically separated? --> LD1 better
      ii. Are classes horizontally separated? --> LD2 better
   e. Plot `ldahist`
   f. Apply `QDA` func
   g. Check differences with LDA
   h. In doubt choose LDA because it's simpler
   i. Standard Error is very high? --> Classes are not well-separated? Try something else

In [ ]: