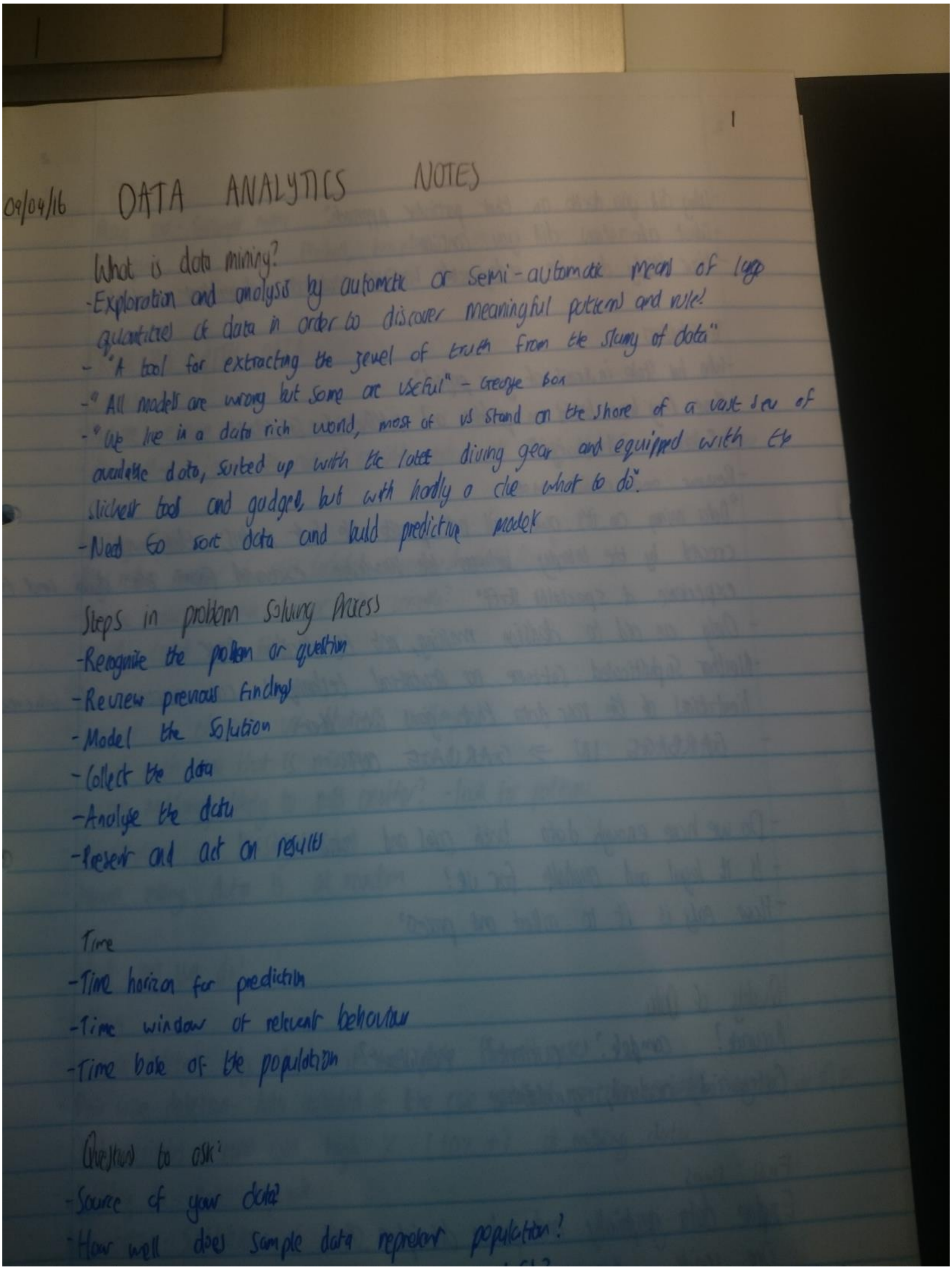


09/04/16 DATA ANALYTICS NOTES

What is data mining?

- Exploration and analysis by automatic or semi-automatic means of large quantities of data in order to discover meaningful patterns and rules
- "A tool for extracting the jewel of truth from the slurry of data"
- "All models are wrong but some are useful" - George Box
- "We live in a data rich world, most of us stand on the shore of a vast sea of available data, suited up with the latest diving gear and equipped with the slickest tools and gadgets but with hardly a clue what to do".
- Need to sort data and build predictive models

Steps in problem solving Process

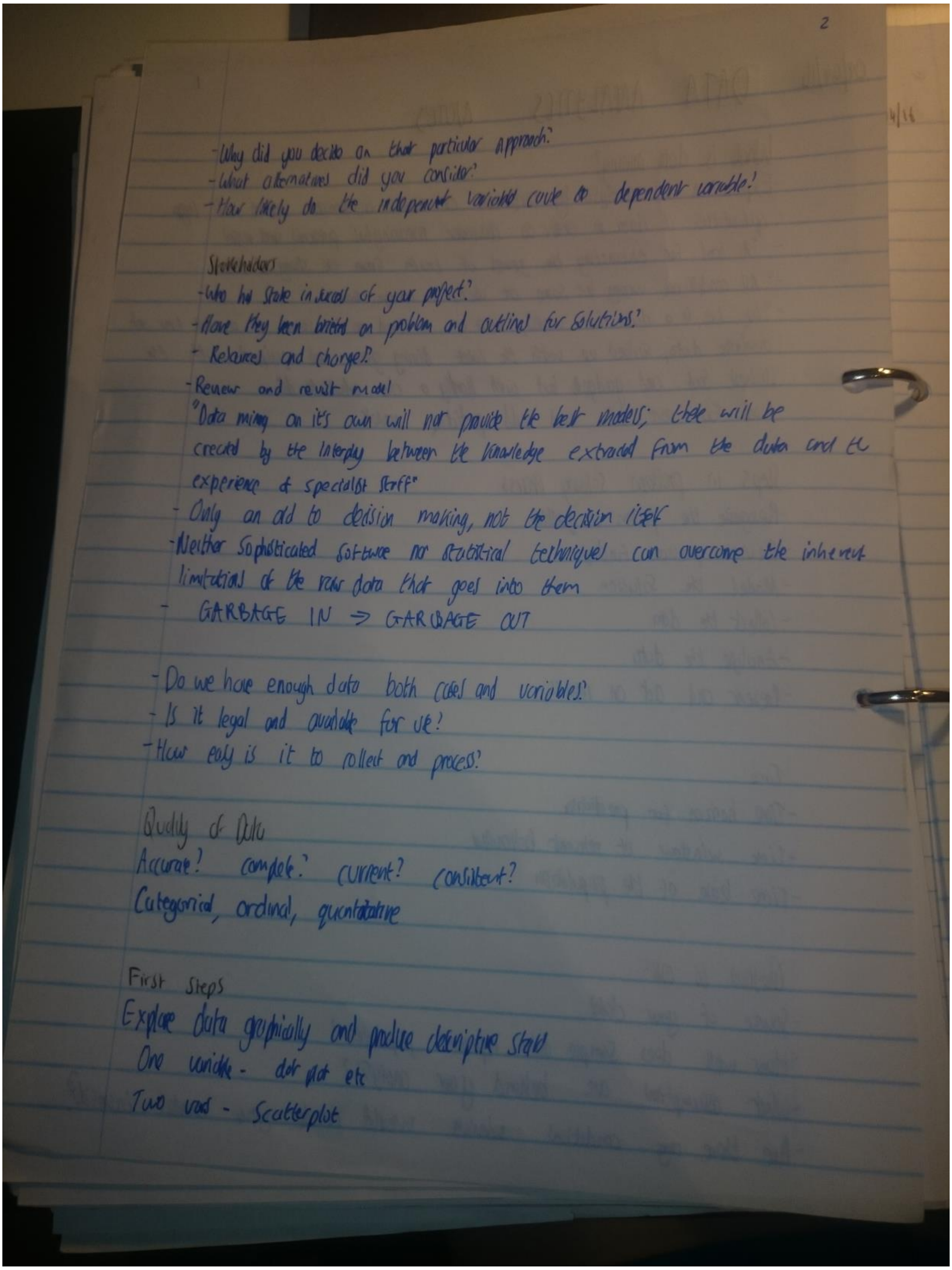
- Recognise the problem or question
- Review previous findings
- Model the Solution
- Collect the data
- Analyse the data
- Present and act on results

Time

- Time horizon for prediction
- Time window of relevant behaviour
- Time base of the population

Questions to ask:

- Source of your data?
- How well does sample data represent population?



- Why did you decide on that particular approach?
- What alternatives did you consider?
- How likely do the independent variable cause to dependent variable!

Stakeholders

- Who has stake in success of your project?
- Have they been briefed on problem and outlines for solutions?
- Resources and changes?
- Renew and revisit model

"Data mining on it's own will not provide the best models; these will be created by the interplay between the knowledge extracted from the data and the experience & specialist staff"

- Only an aid to decision making, not the decision itself
- Neither sophisticated software nor statistical techniques can overcome the inherent limitations of the raw data that goes into them
- GARBAGE IN ⇒ GARBAGE OUT

- Do we have enough data both cases and variables?
- Is it legal and available for use?
- How easy is it to collect and process?

Quality of Data

Accurate? complete? current? consistent?

Categorical, ordinal, quantatative

First steps

Explore data graphically and produce descriptive stats

One variable - dot plot etc

Two vars - Scatterplot

9/04/16 DA

Many var - Scatterplot matrix
One quan, One cat: Piechart, box plot
Two cat: Frequency distribution, tables

PROBLEMS WITH DATA

Outliers

- How are we sure it is an outlier?
- do nothing, ignore the case of variable, replace the values, transform the data

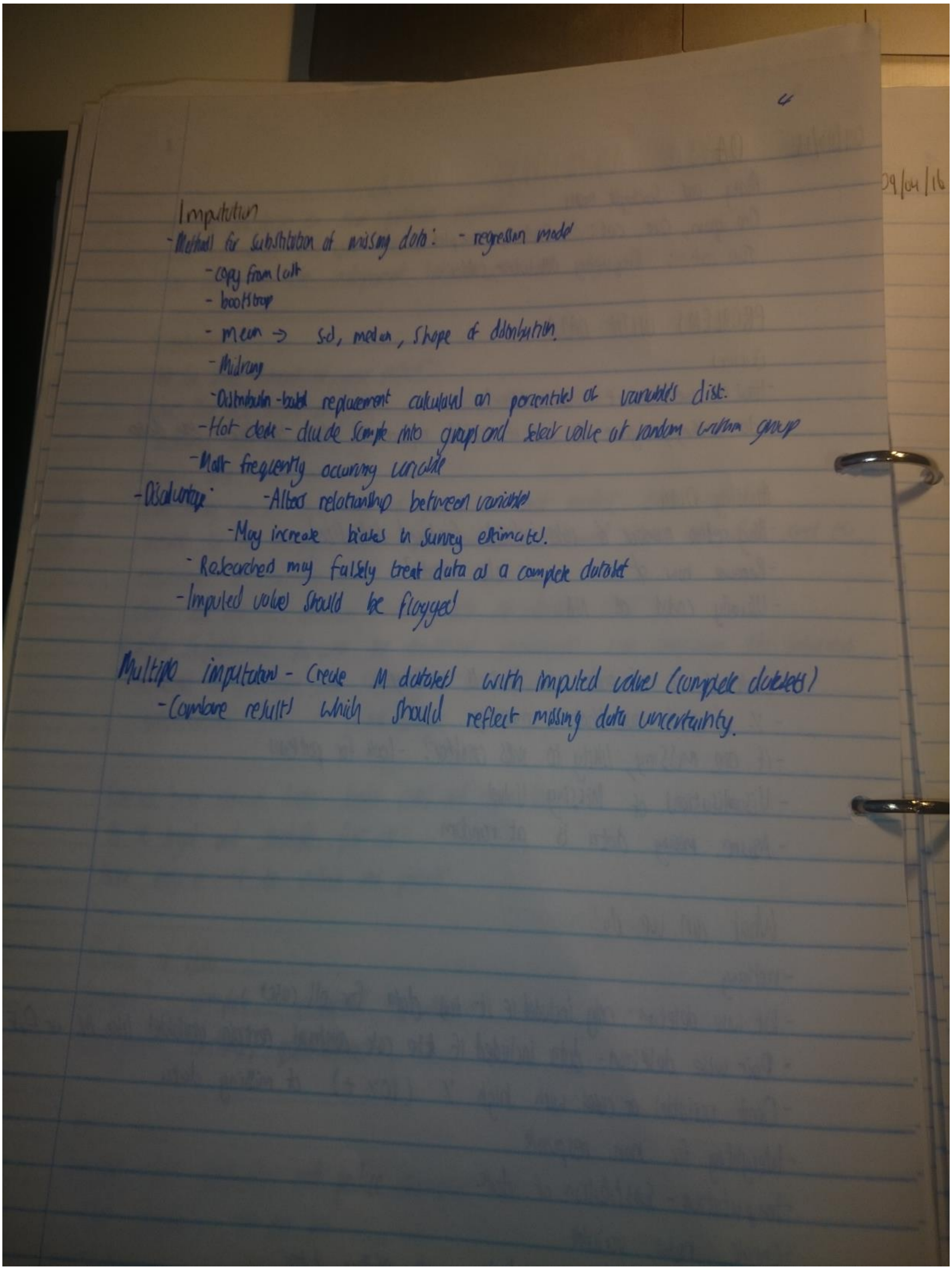
Missing Data

- May reduce number of cases included (reduced data size)
- Remove row of case or else variable?
- Usually coded as NA

- Look at % of each var that is missing
- % of each case that is missing
- if one missing, likely to miss another? - look for patterns
- Visualisations of Missing Values
- Assume missing data is at random

What can we do?

- nothing
- list wise deletion - only included if it has data for all cases
- Pair wise deletion - data included if the case contains certain variables like A,C or D,F
- Omit variables or cases with high % (10% +) of missing data
- Weighting for non response
- Imputation - Substitution of data.
- Create new variable
- Models are unique in dealing with missing data



4

09/04/16

Impututiun

- Methods for substitution of missing data: - regression model
  - copy from (ult
  - bootstrap
  - mean → s.d, median, shape of distribution.
  - Midrang
  - Distribution-based replacement calculated on percentiles of variable's dist.
  - Hot deck - divide sample into groups and select value at random within group
  - Most frequently occuring variable
- Disadvantage: - Alters relationship between variables
  - May increase biases in survey estimates.
  - Researchers may falsely treat data as a complete dataset
  - Imputed values should be flagged

Multiple imputation - Create M datasets with imputed values (complete datasets)
- Combine results which should reflect missing data uncertainty.

09/04/16 DA Classification & REGRESSION TREES

- Supervised learning with a target variable
- Categorical for classification trees, continuous for regression trees
- Need observations on other variables
- Goal: predict or classify an outcome
  - ↳ construct a rule to apply in the future
  - ↳ To see what variables are important/related to target variable

Heart attack example

- 215 cases, 37 died, 100 variables screened

Classification tree

- Upside down tree, root node - all patients
- Two branches defined by a question. Terminal nodes ⇒ classification

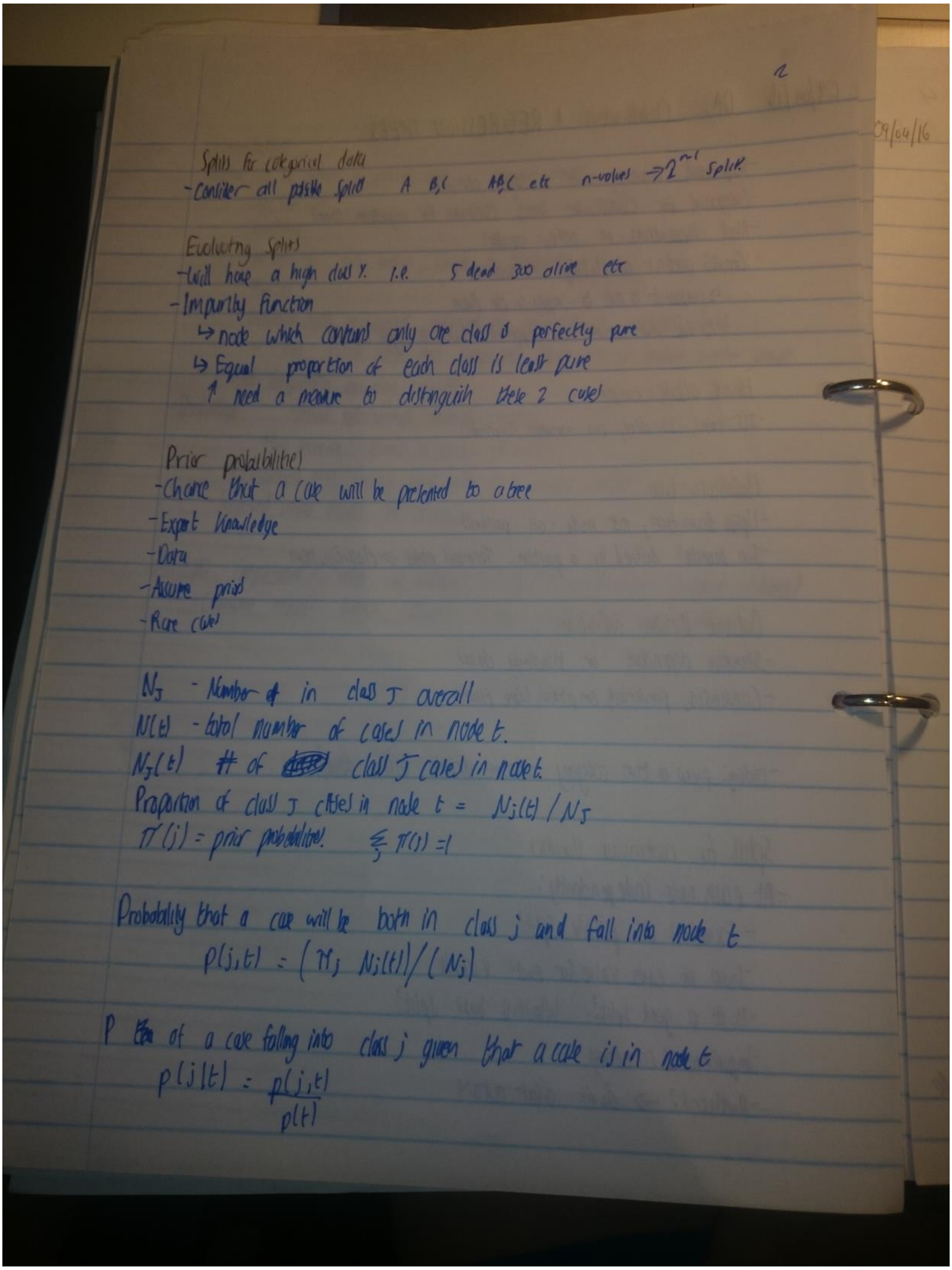
Outcome Variable definition

- Situation dependent or blindingly clear
- Consensus, prescribed, empirical, time frame

- coding: focus on two category outcome variable Yes/No dead/Alive

Splits for continuous Variables

- At each node independantly:
  - Examines all possible splits
  - Looks at each value for each variable
  - Is it a good split? Which is best split?
- Importance of screening
- Outliers? → don't affect outcome



Splits for categorical data

- Consider all possible splits A B,C AB,C etc n-values $\rightarrow 2^{n-1}$ splits

Evaluating splits

- Will have a high class %. i.e. 5 dead 300 alive etc
- Impurity function
  - ↳ node which contains only one class is perfectly pure
  - ↳ Equal proportion of each class is least pure
  - ↑ need a measure to distinguish these 2 cases

Prior probabilities

- Chance that a case will be presented to a tree
- Expert knowledge
- Data
- Assume priors
- Rare cases

N_J - Number of in class J overall

$N(t)$ - total number of cases in node t.

$N_J(t)$ # of class J cases in node t.

Proportion of class J cases in node t = $N_j(t) / N_J$

$\pi(j)$ = prior probabilities. $\sum_j \pi(j) = 1$

Probability that a case will be both in class j and fall into node t

$$p(j,t) = (\pi_j \; N_j(t)) / (N_j)$$

P [illegible] of a case falling into class j given that a case is in node t

$$p(j|t) = \frac{p(j,t)}{p(t)}$$

/04/16 DA CLASSIFICATION & R TREES

P that any case falls into Node t = $p(t) = \sum_j p(j,t)$

Common Impurity Functions

c = # categories in target variable

Entropy = $- \sum_{j=1}^{c} p(j|t) \log_2(p(j|t))$

Gini = $\sum_{i=1}^{c} \sum_{\substack{j=1 \\ j \neq i}}^{c} p(j|t)p(i|t) = 1 - \sum_{j=1}^{c} p^2(j|t)$

$\boxed{\text{When } c=2 \quad \text{Gini} = 2^* p(1|t)p(2|t)}$ Smaller the better → means probability are one big, one small

How to use this to chose split

- look at impurity of parent node t : $i(t)$
- Each split at any node gives us 2 children t_L and t_R
- We measure the impurity of the two children

$i(t) - P_L^* i(t_L) - P_R^* i(t_R)$

P_L = probability going left P_R = Prob going right

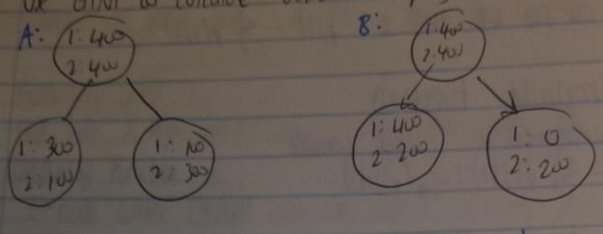
Choosing nodes

- $\Delta(t,s) = i(t) - P_L{}^* i(t_L) - P_R{}^* i(t_R)$
- Smaller values for $i(t_L)$ and $i(t_R)$ are better (higher classification rate)

$P_L + P_R = 1$

- Could be 5000+ splits at each node → evaluate for each one:
- $i(t)$ will be the same for all
- choose split with largest $\Delta(t,s)$
- R calls this improvement.
- R multiplies $\Delta(t,s)$ by N - total no. of cases

Use GINI to calculate decrease in impurity for each split



Improvement : $\Delta(t,s) = i(t) - P_L i(t_L) - P_R i(t_R)$

$i(t) = 2 * p(1|t)\, p(2|t)$

$= 2 * \frac{p(1,t)}{p(t)} \frac{p(2,t)}{p(t)}$

A. $i(t) = 2 * 0.5 * 0.5 = 0.5$

$P_L i(t_L) = \frac{400}{800} * 2 * 0.75 * 0.25 = 0.14$

$P_R i(t_R) = \frac{400}{800} * 2 * 0.25 * 0.75 = 0.19$

$\Delta(t,s) = 0.5 - 0.19 - 0.19 = 0.12.$

B. $i(t) = 2 * 0.5 * 0.5 = 0.5$

$P_L i(t_L) = \frac{6}{8} \times 2 * \frac{2}{3} * \frac{1}{3} = 0.33$

$P_R i(t_R) = \frac{2}{8} * 2 * 0 * \frac{1}{1} = 0$

$\Delta(t,s) = 0.5 - 0.33 - 0 = 0.17$

A has lower impurity / improvement value for the split

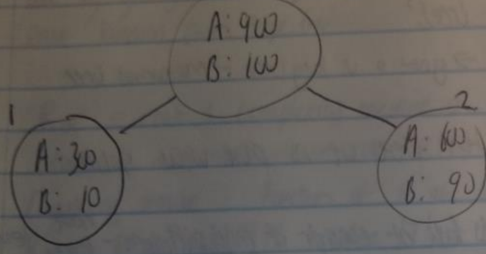
Class Assignment Rule

- Assign a class to every terminal node
- Calculate $p(i|t)$ for each class i
- Let j^* be max of $p(i|t)$
- Assign the node to class j^*
- Can assign a class or output a probability of being assigned to the node.

09/04/16 Classification & R Trees

Priors

- usually use equal priors 0.5, 0.5



$p(i|t)$, $p(i,t)$, $p(t)$ using equal priors and data as priors

Equal priors

$p(A,1) = \frac{\pi_j N_j(t)}{N_j}$ $\quad 0.5 \times \frac{300}{900} = .17 \quad P(\overset{A,2}{\cancel{A,2}}) = 0.5 \times \frac{600}{900} = .33$

$P(\overset{B,1}{\cancel{B,1}}) = 0.5 \times \frac{10}{100} = 0.05 \quad P(\overset{B,2}{\cancel{B,2}}) = 0.5\, \frac{90}{100} = 0.45$

$P(1) = .17 + 0.05 = .22 \quad P(2) = 0.33 + 0.45 = .78$

$P(A|1) = \frac{P(A,1)}{P(1)}\ \frac{.17}{.22} = .77 \quad P(B|1) = \frac{.0.05}{.22} = .42$

~~$P(B|2)$~~ $= P(B|2) .\ \frac{.45}{.71} = .58 \quad P(B|1) = \frac{.05}{.22} = .23$

Assign all cases in node 1 to A, all in node 2 to B.

Data as priors

$\pi_A = 900/1000 = .9 \quad \pi_B = 100/1000 = .1$

$P(A,1) = \pi_j N_j(t)/N_j = 0.9 \times 300/900 = .3 \quad P(A,2) = 0.9\, \frac{600}{900} = .6$

$P(B,1) = .1\,(10/100) = .01 \quad P(B,2) = .1\,(90/100) = 0.09$

$P(1) = .3 + .01 = .31 \quad P(2) = .6 + .09 = .69$

$P(A|1) = {}^{\cdot 3}/_{\cdot 31} = \cdot 97$ $\quad$ $P(A|2) = \frac{\cdot 06}{\cdot 69} = \cdot 87$

$P(B|1) = {}^{\cdot 9}/_{\cdot 31} = 0{\cdot}03$ $\quad$ $P(B|2) = {}^{\cdot 09}/_{\cdot 69} = \cdot 13$

Assign both nodes to A

When to Stop growing trees?

Bottom up - classical CART ⇒ grow a v big tree - maximal tree

- Prune branches

Top down - Stops growing when there are no more ideal splits

Classical CART approach - need to look at concept of misclassification - node / tree level

Confusion matrix ⇒ misclassification rule

		predicted	
		y	n
observed	y		
	N		

Node Misclassification

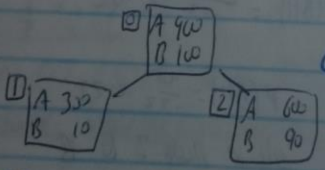
Define $r(t)$ p of misclassification at node t $= 1 - \max(p(i|t))$

$R(t) = r(t) * p(t)$

For the tree we can write:

$R(\hat{\tilde{T}}) = \sum_{\tilde{T}} r(t)p(t)$ where $\tilde{T}$ is all terminal nodes

Using tree:



and data as priors

Node 0: $r(t) = 1 - 0{\cdot}9 = 0{\cdot}1$

$R(t_0) = r(t_0) * p(t_0) = 0.1 * 1 = 0.1$

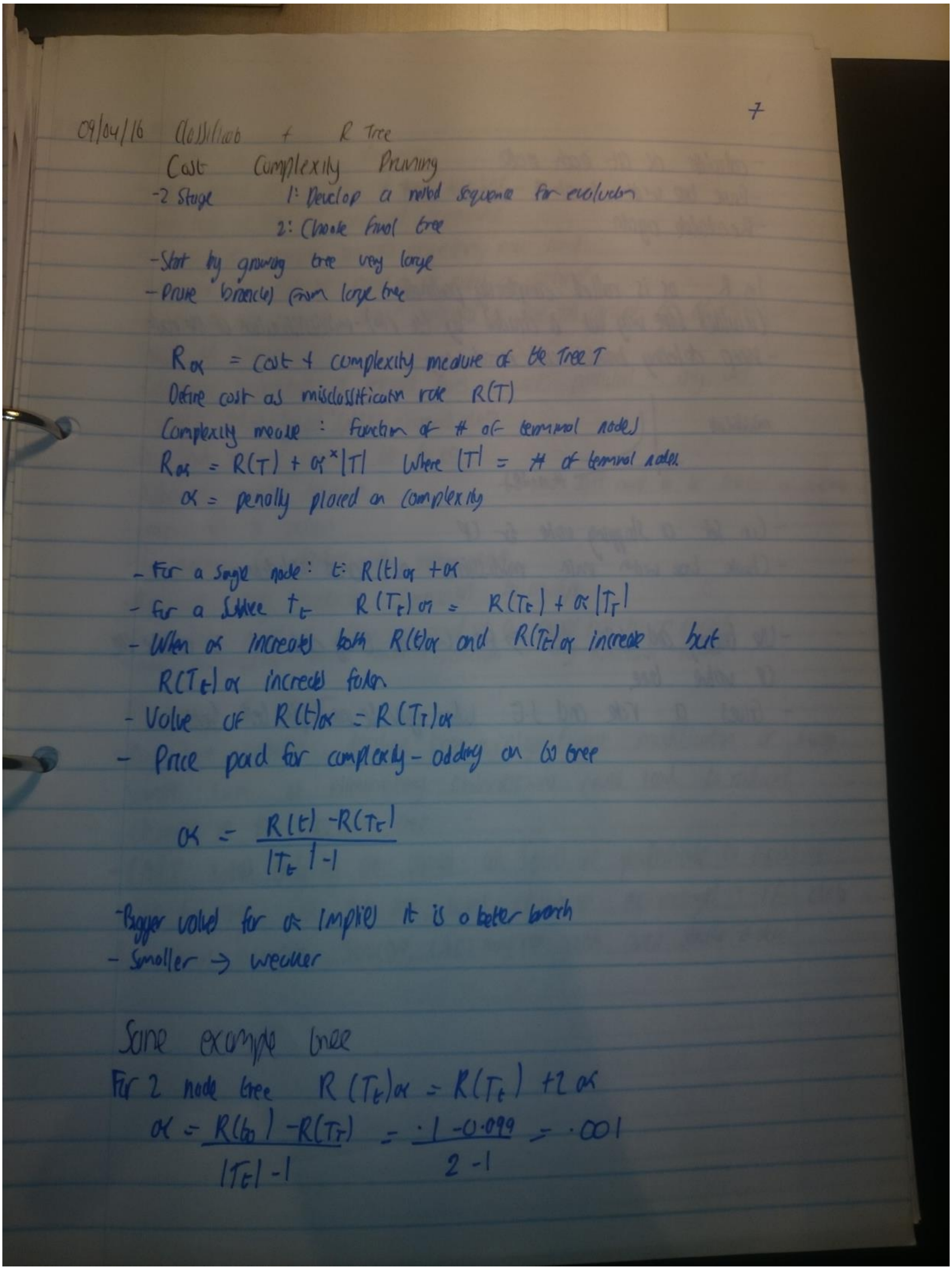
N1: $r(t) = 1 - \cdot 97 = \cdot 03$

$R(t_1) = r(t_1) * p(t_1) = 0{\cdot}3 * \frac{310}{1000} = 0{\cdot}0093$

N(2): $r(t) = 1 - \frac{600}{690} = \cdot 13$

$R(t_2) = r(t_2) * p(t_2) = \cdot 13 * \frac{690}{1000} = 0{\cdot}0897$

$R(T_t) = 0{\cdot}0093 + 0{\cdot}0897 = 0{\cdot}99$



09/04/16 Classification + R Tree

Cost Complexity Pruning

-2 Stage 1: Develop a nested sequence for evaluation

2: Choose final tree

-Start by growing tree very large

-Prune branches from large tree

R_α = cost + complexity measure of the Tree T

Define cost as misclassification rate $R(T)$

Complexity measure : Function of # of terminal nodes

$R_\alpha = R(T) + \alpha^{*}|T|$ Where $|T|$ = # of terminal nodes

α = penalty placed on complexity

- For a single node : t: $R(t)_\alpha + \alpha$
- For a Subtree t_t $R(T_t)_\alpha = R(T_t) + \alpha|T_T|$
- When α increases both $R(t)_\alpha$ and $R(T_t)_\alpha$ increase but $R(T_t)_\alpha$ increases faster
- Value of $R(t)_\alpha = R(T_T)_\alpha$
- Price paid for complexity - adding on to tree

$$\alpha = \frac{R(t) - R(T_t)}{|T_t| - 1}$$

-Bigger values for α implies it is a better branch

- Smaller → weaker

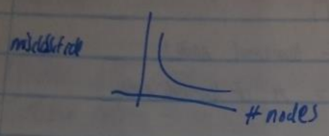
Some example tree

For 2 node tree $R(T_t)_\alpha = R(T_t) + 2\alpha$

$$\alpha = \frac{R(t_0) - R(T_T)}{|T_t| - 1} = \frac{.1 - 0.099}{2 - 1} = .001$$

- calculate α at each node
- Prune tree with lowest α - weakest link
- Recalculate again

In R α is called complexity parameter
Calculated same way but is divided by the r(0) - misclassification of the root
- keep deleting branches and re-calculating



- Can set a stopping value for CP
- Choose tree with min misclassificate or cross validation

- Use training and test data ⇒ see where they cross over ⇒ choose the CP value here.
- Gives a rate and S.E when you use multiple test sets

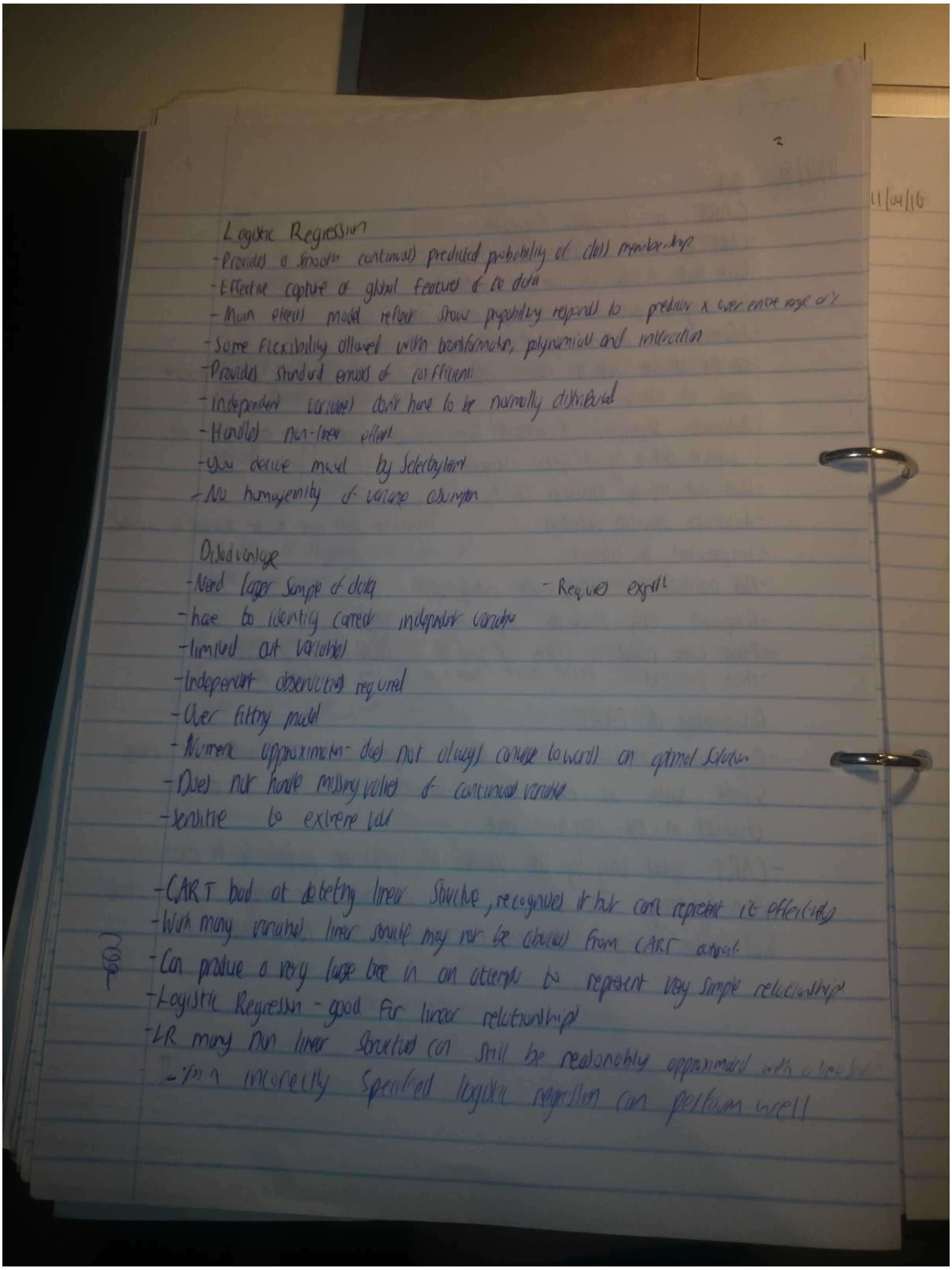
11/04/16 DA

CART vs Logistic Regression

- CART excels in detection of local structure
- Each half of tree is analysed seperately
- Discovery of patterns becomes progressively more local
- Information from different nodes is not pooled or combined
- The fit at one node is never adjusted to take into account fit at another node
- Good at interaction
- Automatic seperation of relevant from irrelevant predictors - only uses one which define "good" splits - Screens data
- Does not require transform like log etc
- Automatic interaction detection - variables don't need to be selected in advance
- Impervious to outliers.
- Has methods for dealing with missing values
- Requires only moderate supervision by analyst
- First time model is often as good as a neural net by an expert
- non parametric - doesn't require specification of any functional form

Disadvantage of CART

- May have unstable decision tree - insignificant modification of learning sample such as eliminating observations could lead to radical changes in the decision tree
- CART splits only by one variable - all splits are perpedicular to axis ie. when splitting all data into boxes, all boxes are rectangles - if data has more complex structure, CART may not catch correct structure of data



Logistic Regression
- Provides a smooth continuous predicted probability of class membership
- Effective capture of global features of the data
- Main effects model reflect how probability responds to predictor x over entire range of x
- Some flexibility allowed with transformation, polynomial and interaction
- Provides standard errors of coefficients
- Independent variables don't have to be normally distributed
- Handles non-linear effects
- You decide model by selecting terms
- No homogeneity of variance assumption

Disadvantage
- Need larger sample of data
- Requires expert
- have to identify correct independent variables
- limited at variables
- Independent observations required
- Over fitting model
- Numeric approximation - does not always converge towards an optimal solution
- Does not handle missing values of continuous variable
- Sensitive to extreme val

- CART bad at detecting linear structure, recognises it but can't represent it effectively
- With many variables, linear structure may not be obvious from CART output
- Can produce a very large tree in an attempt to represent very simple relationships
- Logistic Regression - good for linear relationships
- LR many non linear structures can still be reasonably approximated with a [illegible]
- Even incorrectly specified logistic regression can perform well

11/04/16 DA

Combare CART and Logistic Regression?

- No information left in terminal nodes to support further analysis
- In a well developed CART tree, no other model should be supportable in the nodes
- Run a shallow tree
- Assign every case a terminal node
- Terminal node assignment represented by categorical variable with as many levels as terminal nodes
- Feed this categorical variable in the form of terminal node dummies to LR model.

CART only with j terminal nodes

$y = \beta_0 + \beta_1 N_1 + \ldots + \beta_{j-1} N_{j-1}$

N_j j-1 dummy variables

CART-Logistic Regression Hybrid model

$y = \beta_0 + \beta_1 N_1 + \ldots \beta_{j-1} N_{j-1} + \alpha_1 Z_1 + \ldots \alpha_r Z_r$

Z_i extra variables

= CART node Dummies + Hybrid variables

Surragates

- A variable with possibly equivalent information
- A surrogate is a splitter that splits in a fashion similar to the primary splitter
- Reveals structure of the info in the variables at a particular node in the tree
- If the primary is expensive or difficult to gather - use surrogate instead.
- Use surrogate split if data is missing
- Cases with data for both splits
- For any node t, primary splitter s sends t_L cases to left and t_R to right
- For any other split s^* of the node t into t_L^* and t_R^*
- $N_j(LL)$ number of cases in t that both s and s^* send left for class j.
- $N_j(RR)$ # cases in t that both s and s^* send right for class j.

- Surrogates are node dependant - calculated at the local level.
- Surrogates versus competition
- Useful for examining what node is trying to do
- Can choose how many you calculate at each nod

Variable Importance

- Assess the relative importance of the variables in a tree
- Measured by impurity improvement
- looks at primary Splitter for each node and all the surrogate splits listed on rpart for every node
- Can control the number of Surrogates.
- Calculated over tree
- Calculated as a %

Calculation of $p(S, S^*)$

For this example $j=2$

← from confusion matrix

$NN_{1L} = 500$ $NN_{2L} = 100$ $NN_1 LL =$ [illegible]

$NN^*_{1L} = 400$ $NN^*_{2L} = 150$ $NN_2(LL) = 100$

$NN_{1R} = 100$ $NN_{2R} = 300$ $NN_1(RR) = 50$

$NN^*_{1R} = 150$ $NN^*_{2R} = 250$ $NN_2(RR) = 250$

Calculating Similarity

- Split S and Surrogate Split S^*, for node t.
- j: no. of classes and π_j is prior probability of class j
- Probability S correctly predicts $S^* = p(S, S^*)$

$$p(S,S^*) = P_{RR}(S,S^*) + P_{LL}(S,S^*)$$

$$P_{LL}(S,S^*) = \frac{p(t_L, t_L^*)}{p(t)} = \Sigma_j \frac{\pi_j N_j(LL)}{N_j} / p(t)$$

e.g for $P_{RR}(S,S^*$

11/04/16 DA

$$p(t) = \sum_j p(j,t) = \sum_j \frac{\pi_j N_j(t)}{N_j}$$

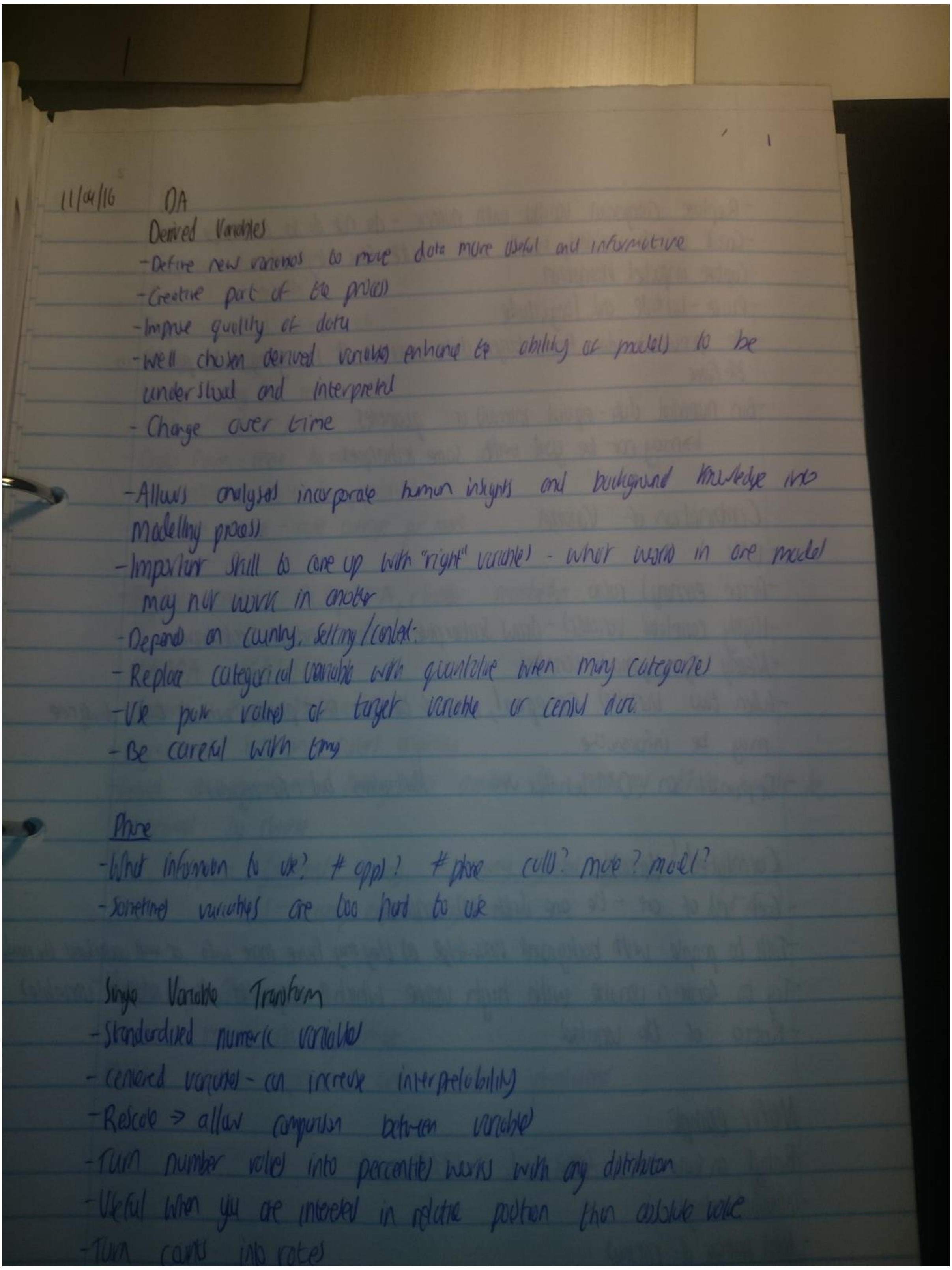
Using data as priors we can show $p(t) = \frac{\sum_j N_j(t)}{\sum_j N_j}$

$\pi_1 = \frac{N_1}{N_1 + N_2} = 0.6 \quad \pi_2 = 0.4$

- In this case, since we are at the root node, $p(t) = 1$
- $P_{LL}(s^*, s) = 0.6\left(\frac{400}{600}\right) + 0.4\left(\frac{100}{400}\right) = 0.5$
- $P_{RR}(s^*, s) = 0.6\left(\frac{200}{600}\right) + 0.4\left(\frac{200}{400}\right) = 0.3$

$p(s, s^*) = P_{LL}(s, s^*) + P_{RR}(s, s^*) = 0.5 + 0.3 = 0.8$

Error $= 1 - 0.8 = 0.2$



11/04/16 DA

Derived Variables

- Define new variables to make data more useful and informative
- Creative part of the process
- Improve quality of data
- Well chosen derived variables enhance the ability of models to be understood and interpreted
- Change over time

- Allows analysts incorporate human insights and background knowledge into modelling process
- Important skill to come up with "right" variables - what works in one model may not work in another
- Depends on country, setting/context.
- Replace categorical variable with quantitative when many categories
- Use past values of target variable or census data
- Be careful with timing

Phone

- What information to use? # apps? # phone calls? make? model?
- Sometimes variables are too hard to use

Single Variable Transform

- Standardised numeric variables
- Centered variables - can increase interpretability
- Rescale ⇒ allow comparison between variables
- Turn number values into percentiles works with any distribution
- Useful when you are interested in relative position than absolute value
- Turn counts into rates

- Replace categorical variables with numeric - do not do so arbitrarily
- Create indicator variables, works well with few categories
- Capture important information
- Place - latitude and longitude
- Use previous data from target data - make sure it is going to be possible in the future
- Bin numerical data - equal intervals or quantiles.
  ↳ may not be good with some techniques

Combination of Variables

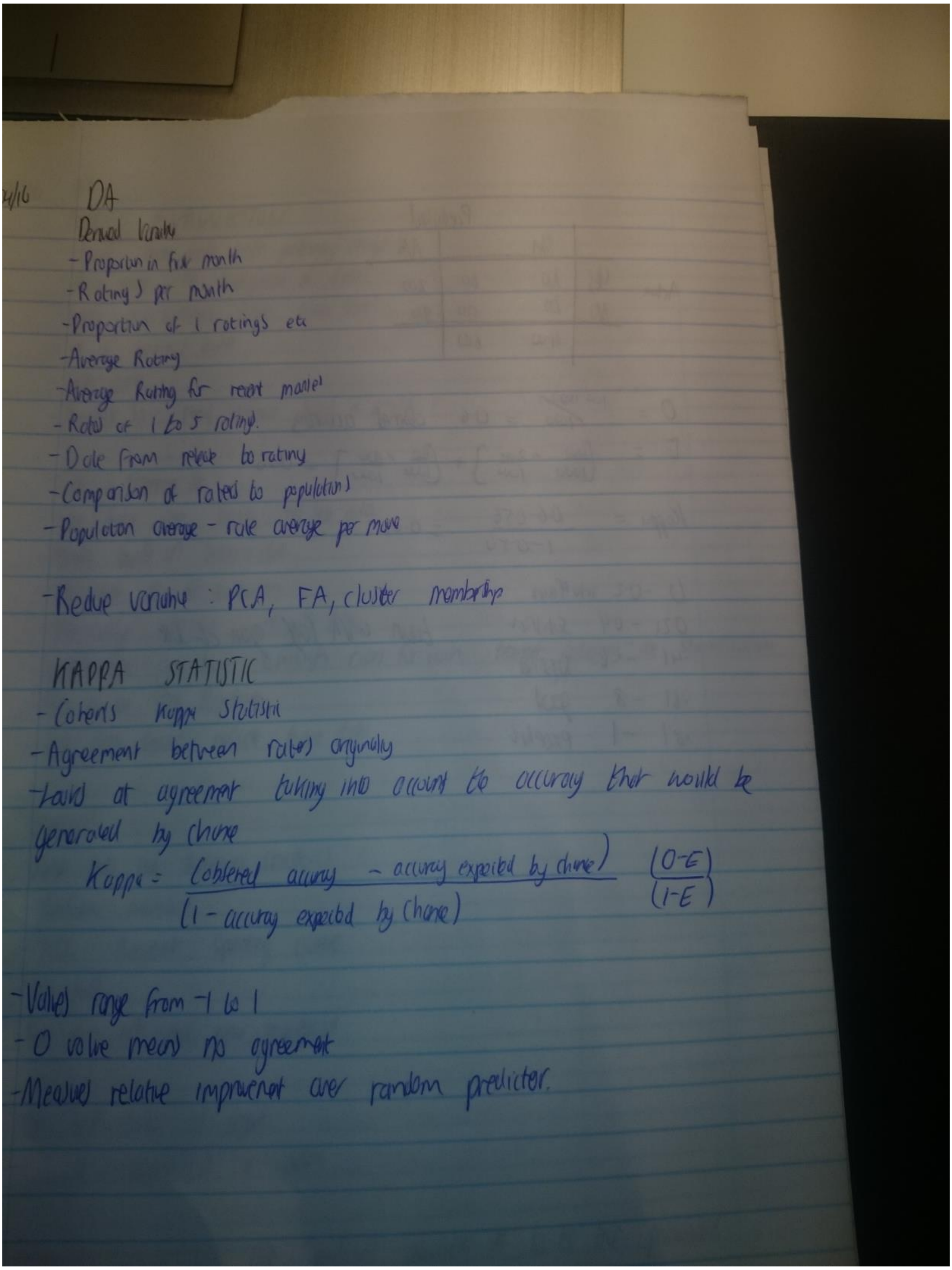
- BMI
- Price earnings ratio
- Highly correlated variables - draw scatterplot before correlation calculation
- Nearly synonymous variables
- When two variables are equal, most of the time few places where the disagree may be informative
- Depends on DM technique used. Multicolinary bad for regression

Correlated Variables

- Get rid of one - the one with less variance
- Talk to people with background knowledge as they may have more info or will understand the model.
- Try to derive a variable with high variance which is independent of its constitute variables
- Ratio of the variables

Netflix example

- Ratings on scale 1-5, 480k users, 17770 movies
- individual ratings
- total number of ratings
- number in first month



4/16 DA

Derived Variables

- Proportion in first month
- Ratings per month
- Proportion of 1 ratings etc
- Average Rating
- Average Rating for recent movies
- Ratio of 1 to 5 ratings.
- Date from release to rating
- Comparison of raters to populations
- Population average - rater average per movie

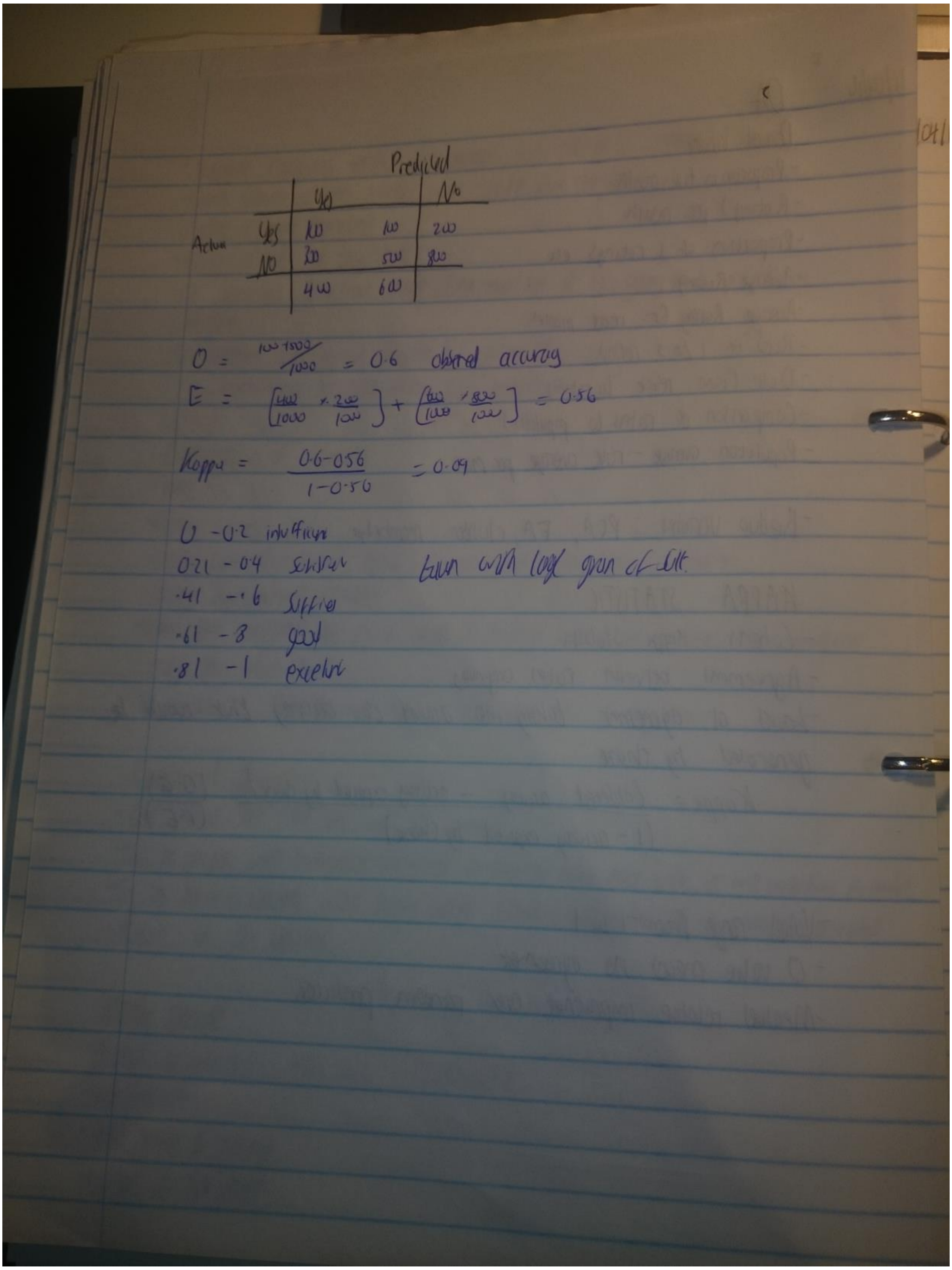
- Reduce variables : PCA, FA, cluster membership

KAPPA STATISTIC

- Cohen's Kappa Statistic
- Agreement between raters originally
- Looks at agreement taking into account the accuracy that would be generated by chance

$$\text{Kappa} = \frac{\text{(observed accuracy} - \text{accuracy expected by chance)}}{\text{(1} - \text{accuracy expected by chance)}} \qquad \frac{(O-E)}{(1-E)}$$

- Values range from -1 to 1
- 0 value means no agreement
- Measures relative improvement over random predictor.



		Predicted		
		Yes		No
Actual	Yes	100	100	200
	No	300	500	800
		400	600	

$O = \frac{100+500}{1000} = 0.6$ observed accuracy

$E = \left[\frac{400}{1000} \times \frac{200}{1000}\right] + \left[\frac{600}{1000} \times \frac{800}{1000}\right] = 0.56$

$$\text{Kappa} = \frac{0.6-056}{1-0.56} = 0.09$$

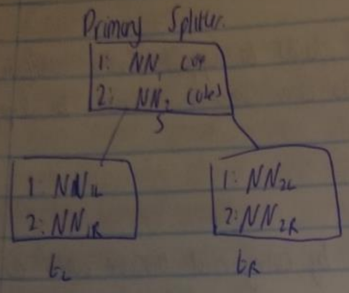
0 -0.2 insufficient

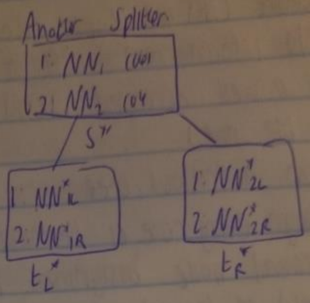
0.21 - 0.4 satisfac   Lower with large gain of [illegible]

.41 -.6 sufficient

.61 -.8 good

.81 -1 excellent



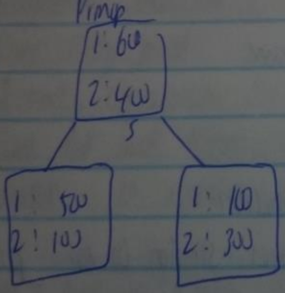


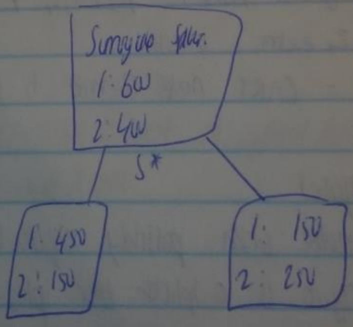
Perfect Splitter.

$NN_{1L} = NN^*_{1L}$ $NN_{2L} = NN^*_{2L}$ $NN_{1R} = NN^*_{1R}$ or $NN_{2R} = NN^*_{2R}$

- Same cols - just not similar names.
- Need to calculate probability that both splitters S and S^* send cases in the same direction
- call this $p(S, S^*)$

EXAMPLE:





Class 1 Primary

Surrog		L	R	
	L	400	50	450
	R	100	50	150
		500	100	

Class 2 Primary

Surr		L	R	
	L	100	50	150
	R	0	250	250
		100	300	

11/04/16 DA

How good a predictor is S^*?

- How well can another split S^* mimic the primary splitter S?
- How high is $p(S,S^*)$?
- Comparison or default rule
- For node t suppose that S sends cases with prob p_L and p_R right (relative)
- New case ⇒ predict t_L if $p_L = \max(p_L, p_R)$ else t_R
- $r(t)$ (misclassification rate) $= \min(p_L, p_R)$

Same example

- Using the data for our priors we can compute these directly from the tree
  $p_L = 0.6$, $p_R = 0.4$
- Default rule ⇒ send everything left, error $= 1 - 0.6 = 0.4$
- Compare this to surrogate rule

Surrogate rule:

- $p(S,S^*) = 0.8$ ⇒ called agreement in R
- error rate $= 0.2$
  ↳ = the table var : $\dfrac{\overset{\text{class 1}}{LL + RR} \quad \overset{\text{class 2}}{+ LL + RR}}{n\ (1000)\ \text{here}}$

Association:

$$= \frac{\min(p_L, p_R) - (1 - p(S^*, S))}{\min(p_L, p_R)}$$

$$= \frac{\text{default mismatch} - \text{surrogate mismatch}}{\text{default mismatch}}$$

$p_L = 0.6$ $p_r = 0.4$ ⇒ $\dfrac{.4(1-0.8)}{.4} = 0.5$ 50% reduction

Relative reduction in error obtained by using S^* to predict ~~S [illegible]~~ S instead of $\max(p_L, p_R)$ called adj in R