London Business School

Data Science

Lecture 4: Classification with Logistic Regression

Kamalini Ramdas

Office Hours

Date & Time	Meeting Link	Meeting ID	Passcode
13/10/21 16:30-17:30	https://zoom.us/j/97129595672	971 2959 5672	523315
20/10/21 16:00-17:00	https://zoom.us/j/91582880144	915 8288 0144	372574

Lectures

Course contents (first part of the course – Kamalini)

- Session 1: The Art & Science of Regression Models For Prediction
- Session 2: More on Using Linear Regression For Prediction
- Session 3: Workshop I – Engineer an algorithm that sets interest rates for new Lending Club loans
  - Group assignment 1, due 6 days after the workshop
- Session 4: Classification using Logistic Regression
- Session 5: Workshop – Invest in a portfolio of Lending Club loans
  - Individual project 1, due 13 days after the end of the workshop

Course contents (second part of the course – Kanishka)

- See canvas syllabus

The classification problem

- Recap from regression
  - Regression analysis is the ***METHOD*** used for finding the linear relationship that links a ***continuous*** variable Y to a set of features (or explanatory variables) X

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots \beta_n X_n + \epsilon$$

  - Features can be continuous or binary dummies (e.g., own house or rent)
- Suppose now the variable Y is a binary variable
  - *Y* is called a binary response variable
  - *Example: Loan paid back or not (charged-off* $Y = 1$*, paid back* $Y = 0$*)*

annual_inc	dti	loan_status
22000	14.29	0
40000	2.55	0
150000	0	0
95000	3.83	1
120000	2.29	0
85000	0.31	0
200000	3.72	1

- Classification problem: Given a set of explanatory variables, is it more likely that an observation will belong to “category” 1 or 0?

Modelling classification

- What is this useful for?
  - **Explanatory**: Quantify the impact of each variable X on the probability that variable Y is equal to 1
  - **Predictive**: Predict the chance that variable Y is equal to 1 using information from the X variables

- Examples
  - Credit card fraud
    - Explanatory: Are jewellery-shop transactions more likely to be fraudulent than average?
    - Predictive: Is a particular transaction fraudulent?
  - Student admissions at LBS
    - Explanatory: Does the GMAT score affect a candidate's probability of being admitted?
    - Predictive: Will I be admitted to LBS?
  - Medicine
    - Explanatory: Does the presence of a specific gene make someone more likely to develop cancer?
    - Predictive: Is John Doe infected by Covid 19?

Today's lecture

- Goals
  - Introduction to the classification problem and logistic regression
  - Discuss how to interpret the results and use them to
    - Reach conclusions regarding explanatory factors
    - Calculate risk scores and make categorical predictions
  - Choose between multiple models
  - Validation using out-of-sample data
  - Provide a "recipe" to follow when you need to do classification using logistic regression
  - Allow you to critically assess the classification work of others

Predicting defaults at the lending club

- **Why is this useful?**

- **How would you start?**

Innovation transforms lending

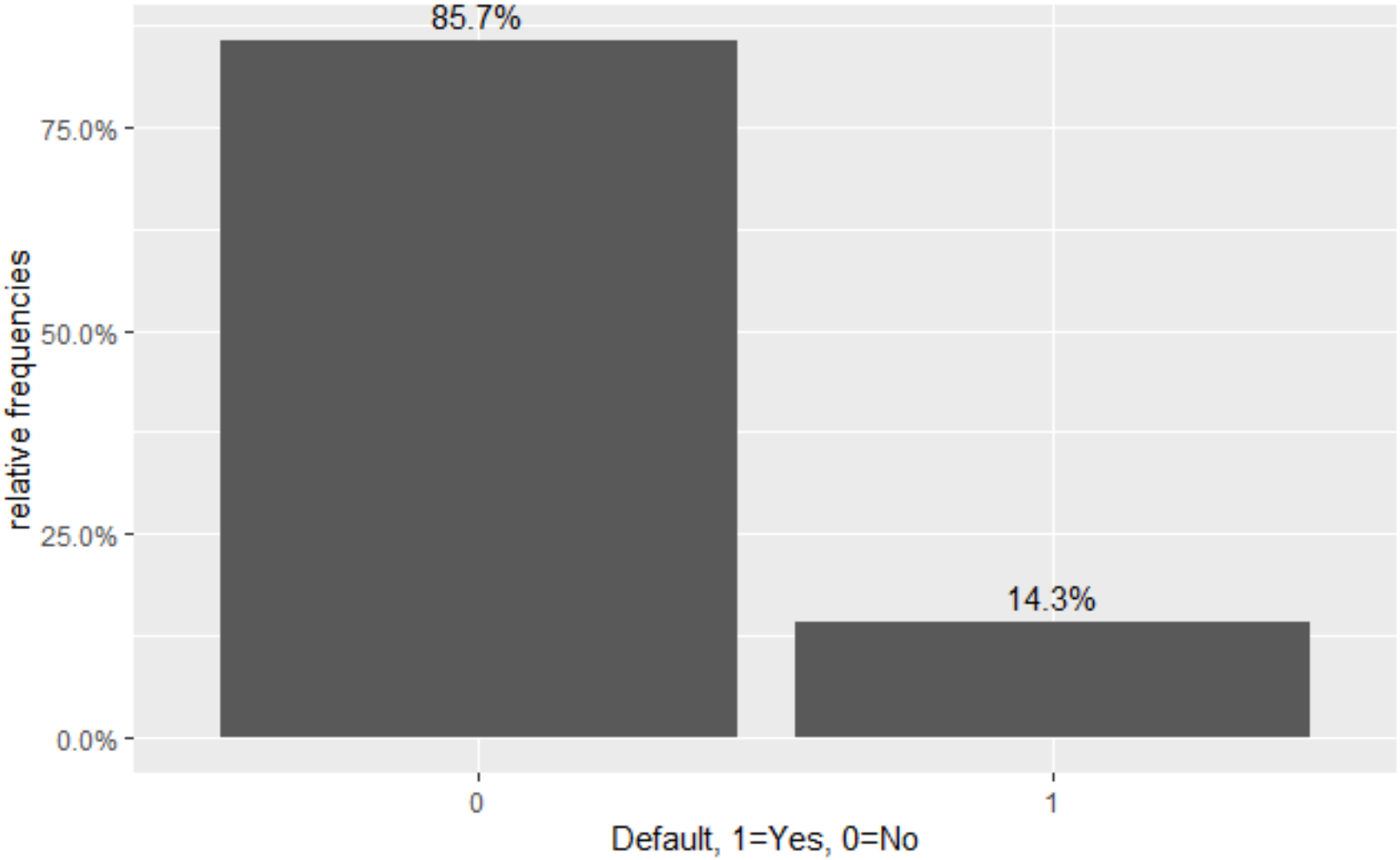
Lending Club is the world's largest marketplace connecting borrowers and investors, where consumers and small business owners lower the cost of their credit and enjoy a better experience than traditional bank lending, and investors earn attractive risk-adjusted returns.[4]

Here's how it works:

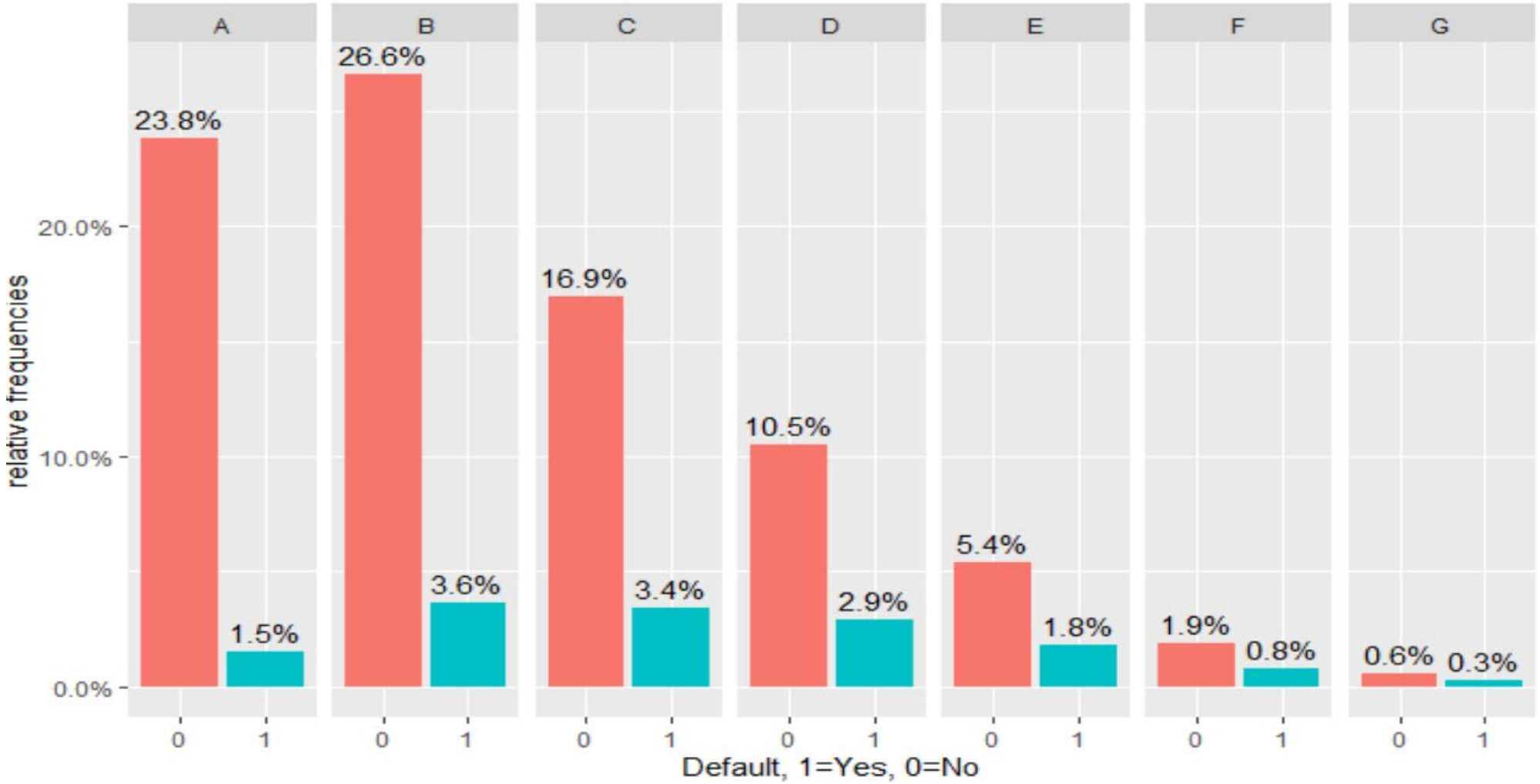
- **Customers interested in a loan complete a simple application at LendingClub.com**
- **We leverage online data and technology to quickly assess risk, determine a credit rating and assign appropriate interest rates. Qualified applicants receive offers in just minutes and can evaluate loan options with no impact to their credit score**
- **Investors ranging from individuals to institutions select loans in which to invest and can earn monthly returns**

The entire process is online, using technology to lower the cost of credit and pass the savings back in the form of lower rates for borrowers and solid returns for investors.

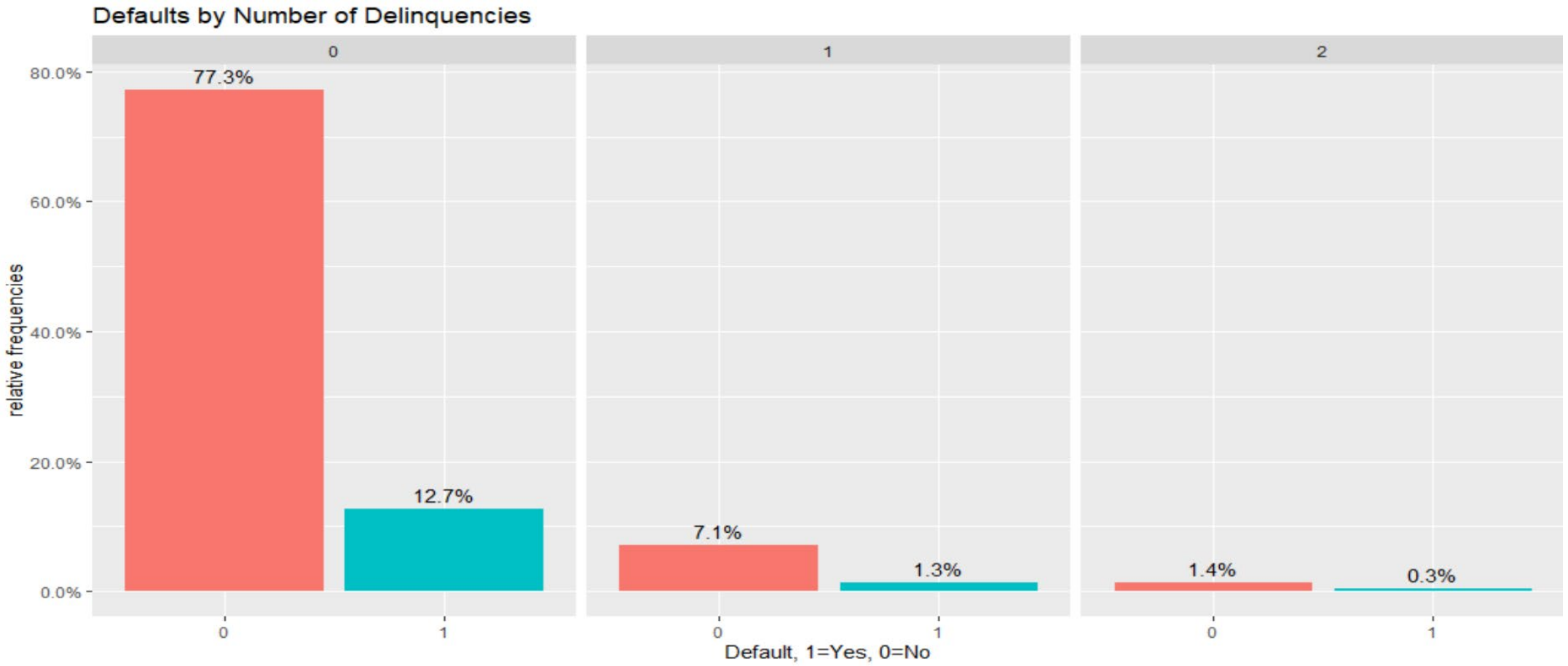
ICE the data!


85.7%
14.3%
75.0%
50.0%
25.0%
0.0%
relative frequencies
0
1
Default, 1=Yes, 0=No

More ICE


A
B
C
D
E
F
G
23.8%
1.5%
26.6%
3.6%
16.9%
3.4%
10.5%
2.9%
5.4%
1.8%
1.9%
0.8%
0.6%
0.3%
20.0%
10.0%
0.0%
relative frequencies
0
1
Default, 1=Yes, 0=No

More ICE


Defaults by Number of Delinquencies
0
1
2
80.0%
60.0%
40.0%
20.0%
0.0%
relative frequencies
77.3%
12.7%
7.1%
1.3%
1.4%
0.3%
0
1
Default, 1=Yes, 0=No

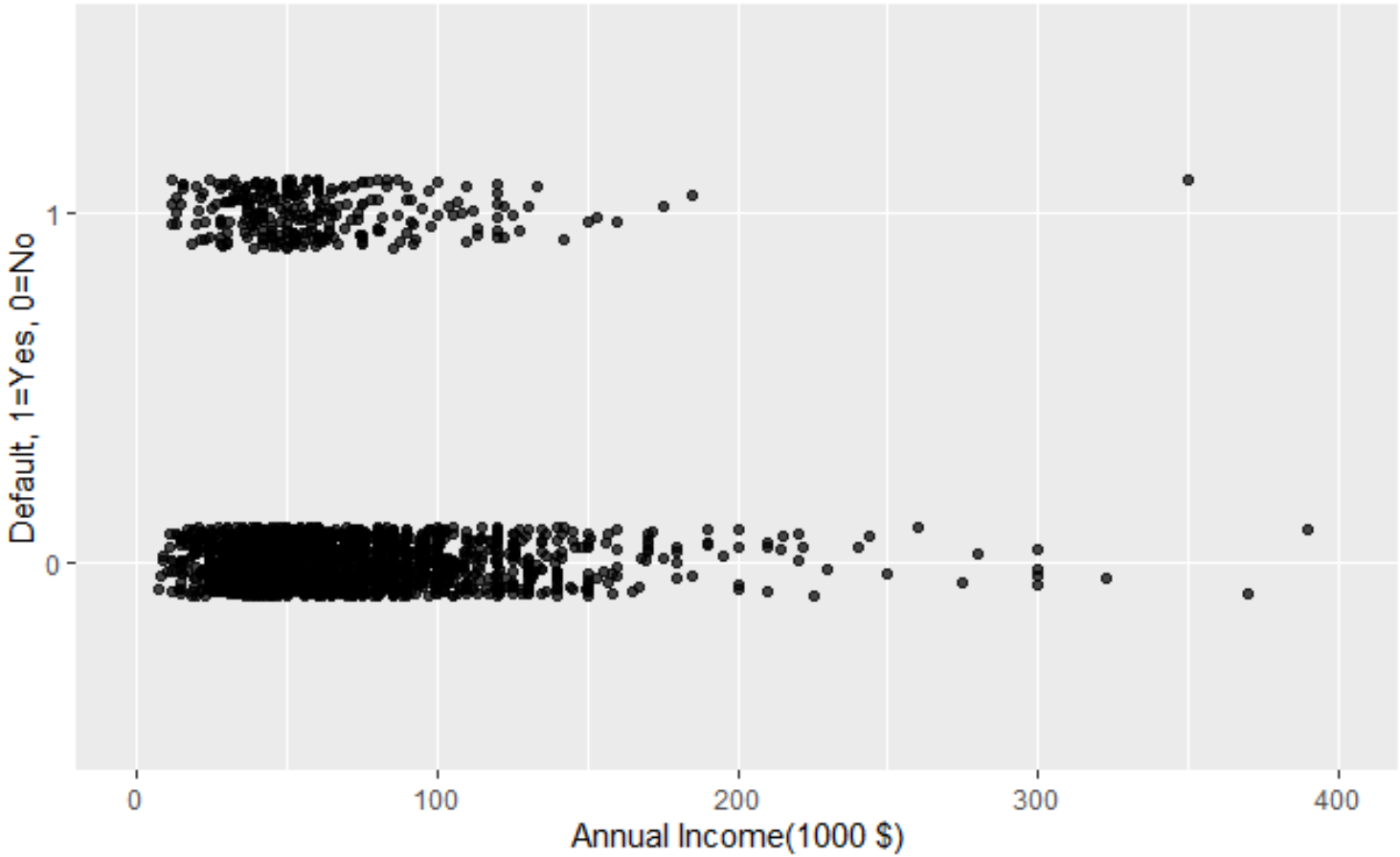
Lending Club and loan defaults

- Let's set $Y = 1$ if charged off and $Y = 0$ otherwise

- Correlation analysis
  - How do you interpret the correlation coefficient?
    - Those have higher income are more likely to pay back
    - Those who borrow more are more likely to default

- Do these relationships sound plausible?

Scatter plot

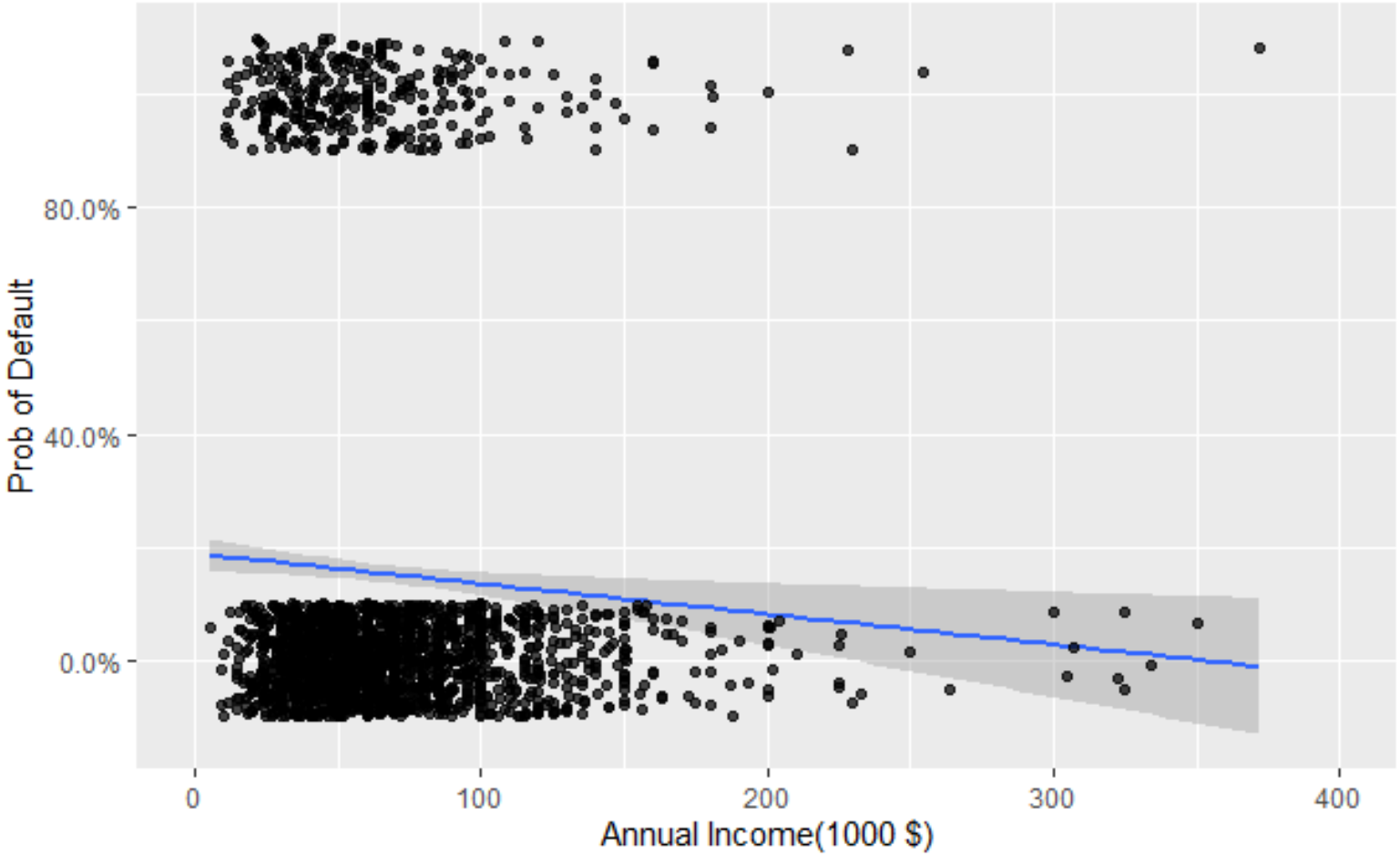


- Can we run a linear regression between the two variables?

$$Default = \beta_0 + \beta_1(Income) + \epsilon$$

- How would you interpret the coefficients?
- Can you see any problem(s) with this regression?

Linear regression plot



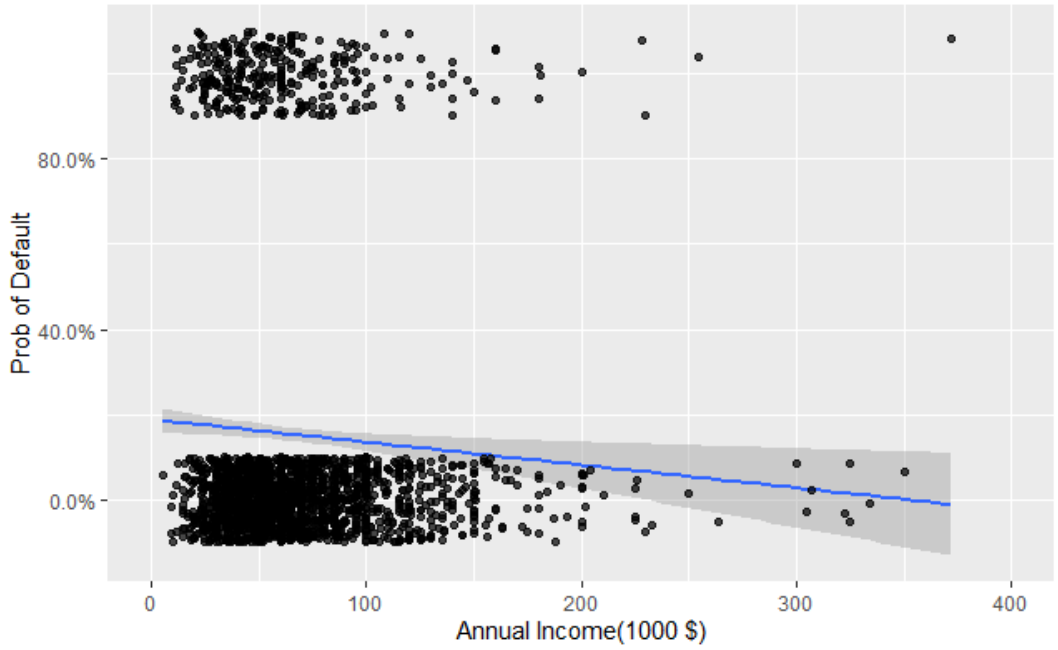
- Can we run a linear regression between the two variables?

$$Default = \beta_0 + \beta_1(Income) + \epsilon$$

- How would you interpret the coefficients?
- Can you see any problem(s) with this regression?

Linear Probability Model

- Running a linear regression is easy to implement… but
- It can be problematic
  - It can predict probabilities that are less than 0 or great than 1
- No micro theory behind it



R-code to generate these figures

```
91   #bar chart of defaults
92   def_vis1<-ggplot(data=lc_clean, aes(x=default)) +geom_bar(aes(y = (..count..)/sum(..count..))) + labs(x="Default,
     1=Yes, 0=No", y="relative frequencies") +scale_y_continuous(labels=scales::percent) +geom_text(aes( label =
     scales::percent((..count..)/sum(..count..) ),y=(..count..)/sum(..count..) ), stat= "count",vjust=-0.5)
93   def_vis1
94
95   #bar chart of defaults per loan grade
96   def_vis2<-ggplot(data=lc_clean, aes(x=default), group=grade) +geom_bar(aes(y = (..count..)/sum(..count..), fill =
     factor(..x..)), stat="count")  + labs(title="Defaults by Grade", x="Default, 1=Yes, 0=No", y="relative frequencies")
     +scale_y_continuous(labels=scales::percent) +facet_grid(~grade) + theme(legend.position = "none") +geom_text(aes( lab
     = scales::percent((..count..)/sum(..count..) ),y=(..count..)/sum(..count..) ), stat= "count",vjust=-0.5)
97   def_vis2
98
99   #bar chart of defaults per number of Delinquencies
100  def_vis3<-lc_clean %>%
101    filter(as.numeric(delinq_2yrs)<4) %>%
102    ggplot(aes(x=default), group=delinq_2yrs) +geom_bar(aes(y = (..count..)/sum(..count..), fill = factor(..x..)),
     stat="count")  + labs(title="Defaults by Number of Delinquencies", x="Default, 1=Yes, 0=No", y="relative frequencies
      +scale_y_continuous(labels=scales::percent) +facet_grid(~delinq_2yrs) + theme(legend.position = "none")
     +geom_text(aes( label = scales::percent((..count..)/sum(..count..) ),y=(..count..)/sum(..count..) ), stat=
     "count",vjust=-0.5)
103
104  def_vis3
106  #scatter plots
107
108  #We select 2000 random loans to display only to make the display less busy.
109  set.seed(1234)
110  reduced<-lc_clean[sample(0:nrow(lc_clean), 2000, replace = FALSE),]%>%
111    mutate(default=as.numeric(default)-1) # also convert default to a numeric {0,1} to make it easier to plot.
112
113
114  # scatter plot of defaults against loan amount
115  def_vis4<-ggplot(data=reduced, aes(y=default,x=I(loan_amnt/1000)))  + labs(y="Default, 1=Yes, 0=No", x="Loan Amnt (1000
     $)") +geom_jitter(width=0, height=0.05, alpha=0.7) #We use jitter to offset the display of defaults/non-defaults to
     make the data easier to interpert. We have also changed the amount to 1000$ to reduce the number of zeros on the
     horizontal axis.
116
117  def_vis4
118
119  #scatter plot of defaults against loan amount.
120  def_vis5<-ggplot(data=reduced, aes(y=default,x=I(annual_inc/1000)))   + labs(y="Default, 1=Yes, 0=No", x="Annual
     Income(1000 $)") +geom_jitter(width=0, height=0.05, alpha=0.7) +  xlim(0,400)
121
122  def_vis5
```

Also look at the rmd file on canvas!

The logistic "random risk" model

- Let U_P denote the "expected risk" of developing a problem – the higher it is the more likely to develop a problem

- We assume that this U_P is linear
  - Given the features $\mathrm{X} = \{\mathrm{X}_1, \mathrm{X}_2, \ldots, X_n\}$ then $U_P|X = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n$

- Actual risk is the expected risk U_P plus a random noise term

- Actual risk and even expected risk U_P is **not observable**
  - We only observe if a problem has occurred or not!

- Given the "expected risk" U_P, the logistic regression model assumes that the probability of observing a problem is given by the logistic function

$$Prob(Default = 1) = \frac{\exp(U_P|X)}{1 + \exp(U_P|X)} = \frac{1}{1 + \exp(-U_P|X)}$$

where, $\exp(n)$ denotes the number $e = 2.71 \ldots$ raised to the power of n

The logistic "random risk" model

- Daniel McFadden shared the 2000 Nobel prize in economics for developing these models
  - Professor at Berkeley

- Some technical notes:
  - Models such as the logistic regression are sometimes called discrete choice models
  - The expected risk factor U_p in discrete choice models is sometimes called Expected Utility, i.e., it represents the expected utility associated with making a specific choice
  - The Logistic regression model assumes that the risk (or the utility) is random and follows a Gumbel distribution with mean U_P
  - If instead of Gumbel distribution we assumed that the risk followed a Normal distribution with mean U_P and variance 1 we would get the ***Probit*** model
    - For most applications, Logistic Regression and Probit regression give almost identical results

The maximum likelihood principle

- We still don't know how to estimate the model coefficients $\beta = \{\beta_0, \beta_1, \ldots, \beta_n\}$
- But, let's assume we think their values are $b_0, b_1, \ldots, b_n$. We can then ask, given these values, what is the likelihood L that the variable Y takes the values observed in my m datapoints:

$$L = \prod_{i=1}^{m} Prob(Y_i | X_i; b)$$

- We can then try to find the values of b that maximize this likelihood L
  - This has parallels to the least-squares estimates of linear regression (but maximizing likelihood is a harder problem than minimizing least squares)
  - We typically maximise the logarithm of the likelihood (log L) – monotone transformation that leads to smaller numbers
  - Besides logistic regression, maximum likelihood has a huge number of applications in statistics

Back to Lending Club

No	loan_status	annual_inc	Probability of default	Prob(Y)
1	0	98	0.020	0.980
2	0	114	0.009	0.991
3	0	72	0.069	0.931
4	0	47.5	0.202	0.798
5	0	122.748	0.006	0.994
6	1	60	0.119	0.119
7	1	24	0.450	0.450
8	0	18	0.525	0.475
9	1	60	0.119	0.119
10	0	95.004	0.023	0.977

- Expected risk: $U_P = \beta_0 + \beta_1 Annual\ Income$
  - Let's assume $b_0 = 1$ and $b_1 = -0.05$
- For loan 1:
  - U_P estimate $= b_0 + b_1 \times 98 = 1 - 0.5 \times 98 = -3.9$
  - Estimated probability of default $p_1 = \frac{1}{1+\exp(3.9)} = 0.02\%$
- Similarly, we can calculate the other loans' probabilities $(p_2, p_3, \dots, p_{10})$
- The Likelihood of having defaults for loans (6, 7 and 9) and not in any of the other defaults

$$L = (1-p_1) \times (1-p_2) \times (1-p_3) \times (1-p_4) \times (1-p_5) \times p_6 \times p_7 \times (1-p_8) \times p_9 \times (1-p_{10})$$

- The maximum likelihood principle asks what values of b_0, b_1 maximize L?
  - Need to use a numerical solver to maximize the likelihood (or the logarithm of the likelihood)
  - Unlike linear regression it's not possible to estimate logistic regression using linear algebra

Logistic regression

```
logistic1<-glm(default~I(annual_inc/1000), family="binomial", lc_clean)
summary(logistic1)
```

```
Call:
glm(formula = default ~ I(annual_inc/1000), family = "binomial",
    data = lc_clean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6246  -0.5786  -0.5572  -0.5115   3.6388

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.5191545  0.0292430  -51.95   <2e-16 ***
I(annual_inc/1000) -0.0040801  0.0003998  -10.21   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31130  on 37868  degrees of freedom
Residual deviance: 31005  on 37867  degrees of freedom
AIC: 31009

Number of Fisher Scoring iterations: 5
```

- Using maximum likelihood we can estimate the coefficients that are most likely to have generated our data
  - I have done so here using all of the data & I have transformed annual income to be in $ 1000’s
- The risk model is
- $U = -1.51915 - .00408$ * Income ($1000s)
  - What is the probability of default for a loan applicant with income 100 (000 USD)

$$P = 1/(1 + \exp(1.51915 + 0.00408 \times 100)) = 12.7\%$$

Logistic regression

```
Call:
glm(formula = default ~ I(annual_inc/1000), family = "binomial",
    data = lc_clean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6246  -0.5786  -0.5572  -0.5115   3.6388

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.5191545  0.0292430  -51.95   <2e-16 ***
I(annual_inc/1000) -0.0040801  0.0003998  -10.21   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31130  on 37868  degrees of freedom
Residual deviance: 31005  on 37867  degrees of freedom
AIC: 31009

Number of Fisher Scoring iterations: 5
```

- The std error and p-value have a similar interpretation as the linear regression
  - To assess coefficient significance at the 5% level we look at the estimated standard error and p value the usual way
- Coefficients no longer have a ***Marginal Value*** interpretation
  - In a linear model, the coefficient β_1 had a marginal effect interpretation: An increase in variable X by 1 unit increases variable Y by 1 unit.
  - This in not the case in logistic regression because it’s a non-linear model
  - Using elementary calculus, the marginal effect of variable X is equal to $\beta\ p\ (1-p)$.
    - Therefore, the marginal effect will be different for different values of *X*. Largest effect for $p = \frac{1}{2}$
    - But it has the same sign as the coefficient β

Goodness-of-Fit

- In linear regression we assess in-sample goodness-of-fit by using the R-Squared
- The equivalent notion in logistic regression is deviance – we want deviance to be as small as possible
  - $\text{Deviance} = -2\log(L)$ where L is the maximum likelihood
- R reports the deviance of a model with only the intercept (Null Deviance) and the deviance of the model with the features
- AIC is similar to deviance but penalizes for number of coefficients (like adj-R^2 in linear regression)
  - $AIC = -2\log(L) + 2k$, k is the number of estimated coefficients
  - Useful for comparing different models with different number of features estimated on the same dataset (in-sample comparison)

```
Call:
glm(formula = default ~ I(annual_inc/1000), family = "binomial",
    data = lc_clean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6246  -0.5786  -0.5572  -0.5115   3.6388

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.5191545  0.0292430  -51.95   <2e-16 ***
I(annual_inc/1000) -0.0040801  0.0003998  -10.21   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31130  on 37868  degrees of freedom
Residual deviance: 31005  on 37867  degrees of freedom
AIC: 31009

Number of Fisher Scoring iterations: 5
```

Feature Engineering

- **Same principles as linear regression**
  - Features: ***Term, income, dti, grade***, number of delinquencies, employment length, etc
    - Some of these are numerical others are factors. How do we use factor variables?
  - Interaction terms: *Perhaps the loan amount affects 36-month loans differently than 60-month loans*. How do you model this?
    - Interactions between two factor features, a factor and a numerical feature, two numerical features
  - Non-linear terms: *Perhaps a small increase in the loan amount doesn't affect interest rate so much but a large increase does.* How would you model this?
    - ***Polynomial terms*** (powers of a feature) or any ***other non-linear transformation*** (better have a good reason for the non-linear transformation)
    - Dummy variable creation → converting a numerical variable into a factor variable (e.g., low, mid, high income, or ***deciles of income***). This is a non-parametric way of modelling non-linear relationships
- **Look for data outside your model**
- **Feature engineering is more of an art than science! Know your context (or work with people who do)!**

Multivariate logistic regression

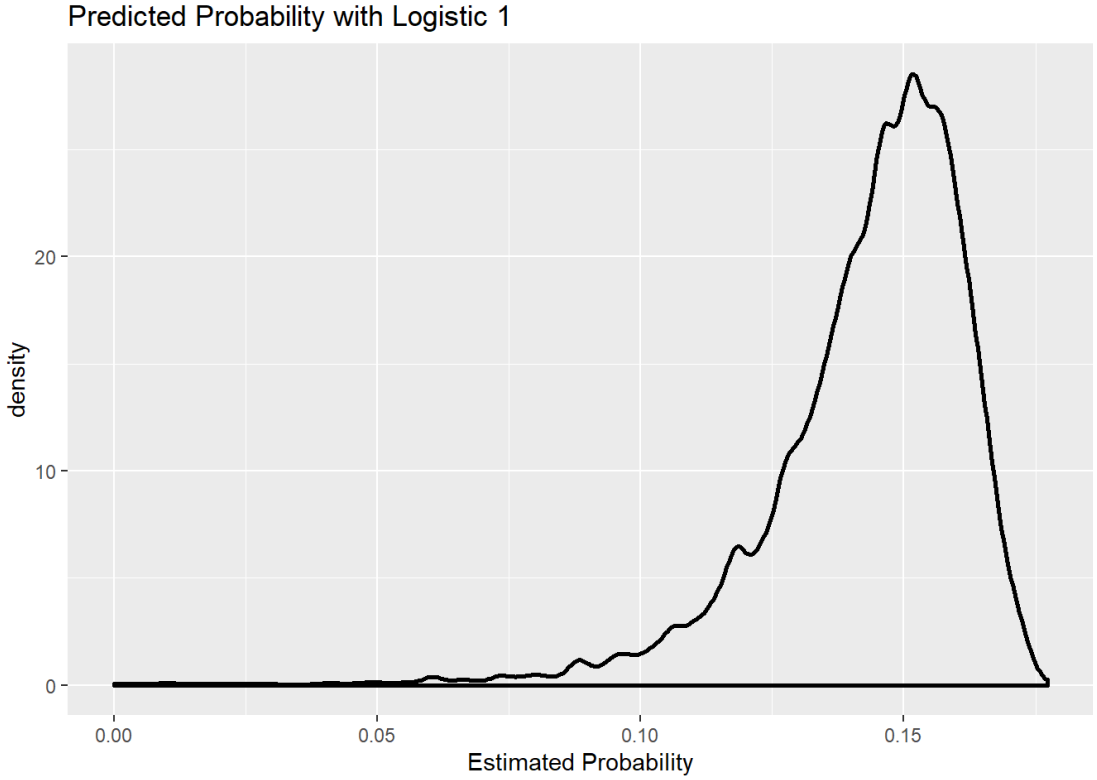
Logistic 1

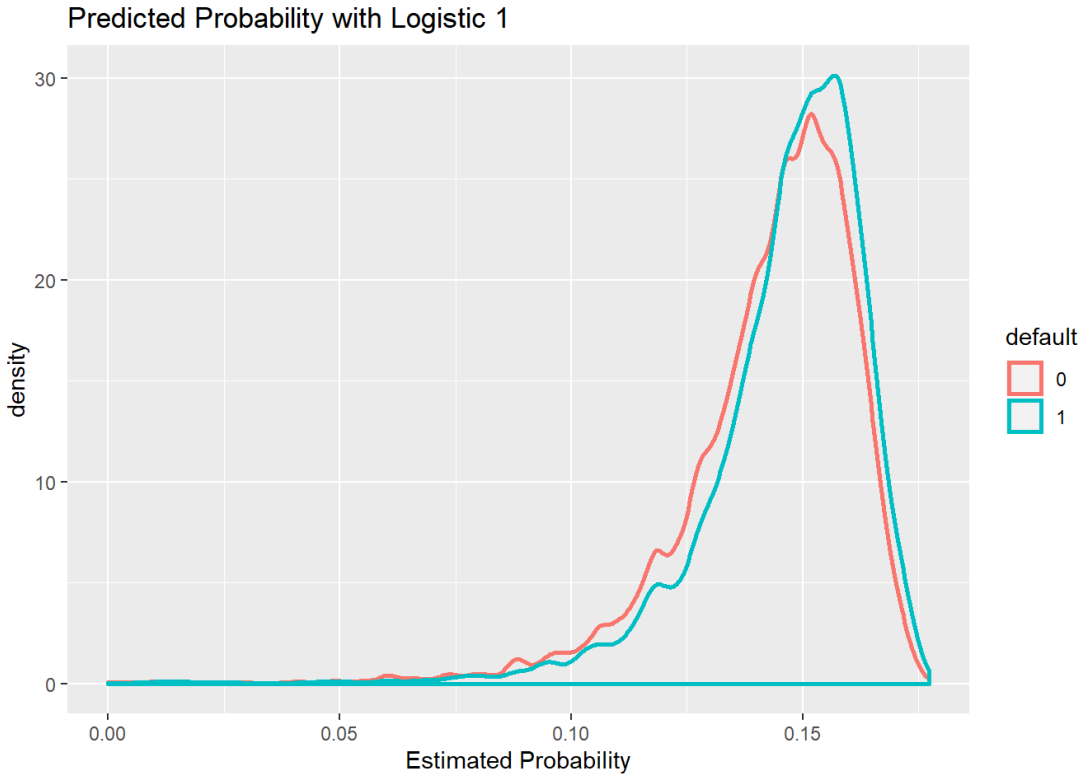
```
Call:
glm(formula = default ~ I(annual_inc/1000), family = "binomial",
    data = lc_clean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.6246  -0.5786  -0.5572  -0.5115   3.6388

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.5191545  0.0292430  -51.95   <2e-16 ***
I(annual_inc/1000) -0.0040801  0.0003998  -10.21   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31130  on 37868  degrees of freedom
Residual deviance: 31005  on 37867  degrees of freedom
AIC: 31009

Number of Fisher Scoring iterations: 5
```

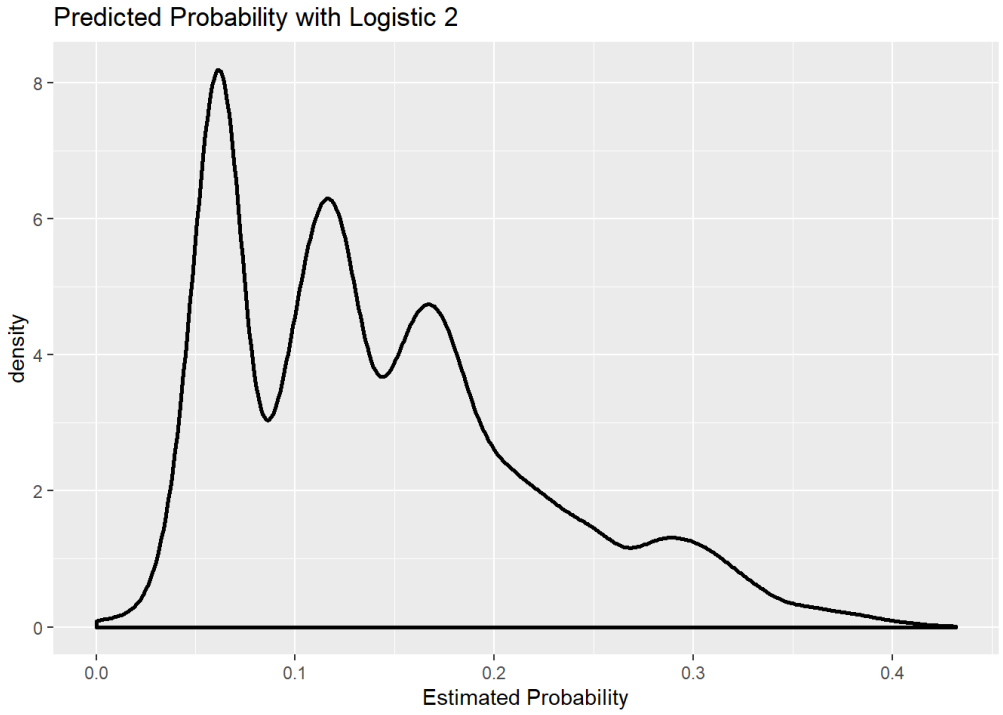
Logistic 2

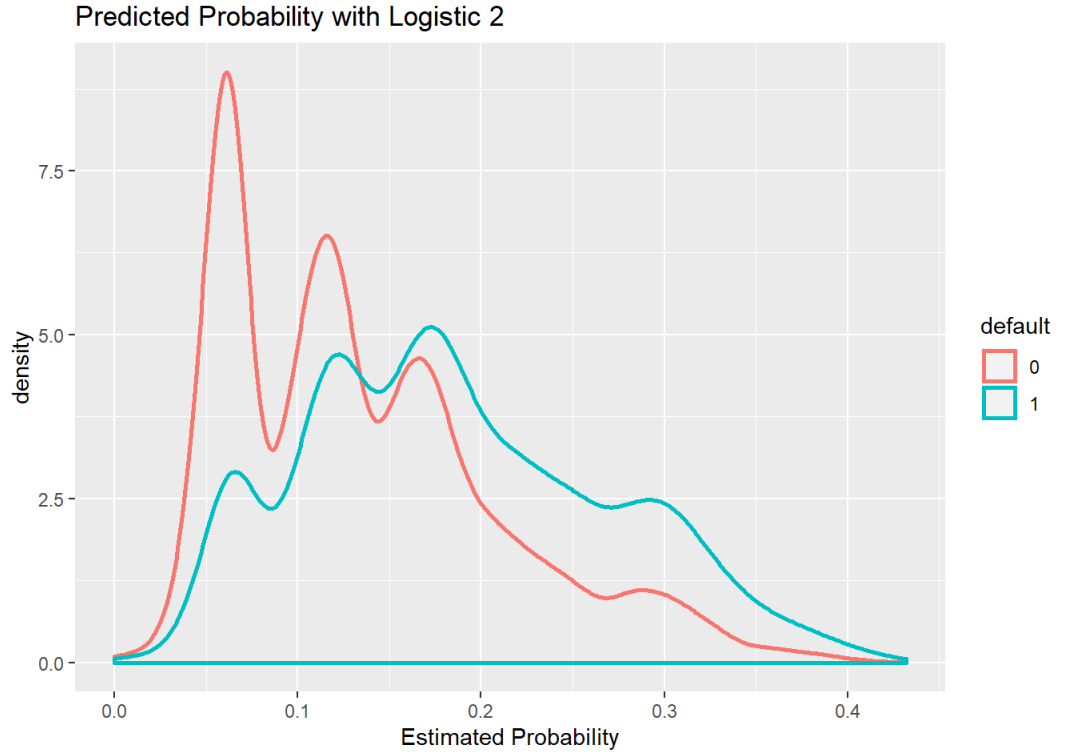
```
Call:
glm(formula = default ~ annual_inc + term + grade + loan_amnt,
    family = "binomial", data = lc_clean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.0635  -0.6079  -0.4815  -0.3455   4.0988

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.432e+00  5.056e-02 -48.093   <2e-16 ***
annual_inc  -6.019e-06  4.727e-07 -12.733   <2e-16 ***
term60       4.790e-01  3.561e-02  13.451   <2e-16 ***
gradeB       6.599e-01  5.270e-02  12.521   <2e-16 ***
gradeC       1.031e+00  5.392e-02  19.128   <2e-16 ***
gradeD       1.288e+00  5.703e-02  22.578   <2e-16 ***
gradeE       1.415e+00  6.662e-02  21.241   <2e-16 ***
gradeF       1.666e+00  8.630e-02  19.302   <2e-16 ***
gradeG       1.769e+00  1.340e-01  13.198   <2e-16 ***
loan_amnt    2.841e-06  2.347e-06   1.211    0.226
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 31130  on 37868  degrees of freedom
Residual deviance: 29286  on 37859  degrees of freedom
AIC: 29306

Number of Fisher Scoring iterations: 5
```

Which is better, logistic 1 or logistic 2?


Predicted Probability with Logistic 1
density
Estimated Probability
0
10
20
0.00
0.05
0.10
0.15


Predicted Probability with Logistic 1
density
Estimated Probability
default
0
1
0
10
20
30
0.00
0.05
0.10
0.15


Predicted Probability with Logistic 2
density
Estimated Probability
0
2
4
6
8
0.0
0.1
0.2
0.3
0.4


Predicted Probability with Logistic 2
density
Estimated Probability
default
0
1
0.0
2.5
5.0
7.5
0.0
0.1
0.2
0.3
0.4

Topics to follow

- Logistic regression is not classification. It's a probability estimate
  - How do we make use of this model to make **categorical** predictions?
  - How can we assess how well the model performs?
  - How do we compare performance of different models?

  - These are more difficult questions in classification problems than linear regression. But not impossible!

Making predictions: From probability to classification

- Logistic regression (or any other classification model for that matter) gives you a probability estimate (sometimes called score)
  - This is a number between 0 and 1
  - Intuitively, this is a number proportional to the chances of observing “success”
- But in classification problems we are interested in predicting outcomes: 0 or 1
- **We need to convert these scores (number between 0-1) to predictions (“success” of “failure”)**

Making predictions: From probability to classification

- **Converting scores to classification**
  - For each data point we can estimate the probability $Y = 1$
  - Choose a cut-off point, say 0.25
  - If the predicted probability is greater than the cutoff (>0.25) we predict that Y will take the value 1, otherwise zero

- **Some of these predictions will be right and some wrong**
  - Confusion matrix records the actual vs predicted outcomes

		Prediction	
		Good loan	Bad loan
Actual	Good loan	29,445	4,131
	Bad loan	4,131	1,298

Assessing model performance

- **Consider the following scenario:**
  - 1 in 1,000,000 credit card transactions are fraud.
  - My classifier estimates all the transactions as non-fraud so we can compute its confusion matrix for a representative data as follows

		Prediction	
		Good loan	Bad loan
Actual	Good loan	999,999	0
	Bad loan	1	0

  - What is the accuracy? 99.9999%
  - Obviously, this is misleading
  - So besides accuracy, specificity and sensitivity are also important!
    - Specificity= Prob of predicting 0 if actual is 0 = 100%
    - Sensitivity=Prob of predicting 1 if actual is 1 = 0%
  - Through the choice of cut-off we can always improve sensitivity at the expense of specificity and vice versa
  - Deciding on the cut-off value will depend on the relative cost of the two types of errors

Assessing model performance

- **Language of classification: Here is the common language we use to indicate different parts of the confusion matrix**
  - Usually, we set the value of the outcome we are interested in equal to 1

	Reference				Reference	
Prediction	0	1		Prediction	0	1
0	29445	4131		0	True Negative	False Negative
1	2995	1298		1	False Positive	True Positive

- **Measure I: (Plain accuracy)**

$$Accuracy = \frac{Number\ of\ correct\ decisions\ made}{Total\ number\ of\ decisions\ made}$$

What is the accuracy of this model =(29445 + 1298)/(37869)=81.18%

Confusion Matrix Statistics

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 29445  4131
         1  2995  1298

               Accuracy : 0.8118
                 95% CI : (0.8079, 0.8158)
    No Information Rate : 0.8566
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1608

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.23909
            Specificity : 0.90768
         Pos Pred Value : 0.30235
         Neg Pred Value : 0.87697
             Prevalence : 0.14336
         Detection Rate : 0.03428
   Detection Prevalence : 0.11336
      Balanced Accuracy : 0.57338

       'Positive' Class : 1
```

- **Measure I:**
  - **Accuracy**: How often is the model right
  - No information rate: The accuracy of a classifier do without any information
    - Can you guess the classifier has 85.66% accuracy?
- **Measure II:**
  - **Sensitivity**: True positive rate (i.e., what proportion of bad loans did we catch?)
  - **Specificity**: True negative rate (i.e., what proportion of good loans did we identify as good loans)
    - Can you guess what would the sensitivity and specificity of the maximum accuracy no-information classifier be?

Confusion Matrix Statistics

```
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 29445  4131
         1  2995  1298

               Accuracy : 0.8118
                 95% CI : (0.8079, 0.8158)
    No Information Rate : 0.8566
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1608

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.23909
            Specificity : 0.90768
         Pos Pred Value : 0.30235
         Neg Pred Value : 0.87697
             Prevalence : 0.14336
         Detection Rate : 0.03428
   Detection Prevalence : 0.11336
      Balanced Accuracy : 0.57338

       'Positive' Class : 1
```

- Is the model better than random guess?
  - Evaluating goodness of fit using the confusion matrix is a good first step
  - One problem is that it depends on the cut-off value
    - Is poor performance because of bad fit or poor choice of cut-off?
- We need sensible cutoff threshold!

Choosing the cut-off
Expected value approach

- One way of choosing the cut-off value is by estimating the expected cost associated with decisions
- For example:
  - We will only invest in loans that are predicted not to default (i.e., prediction 0)
  - For each loan that does not default (true negative) we make a profit of $10
  - For each loan that defaults (false negative) we lose $70

Confusion Matrix				Cost Matrix		
	Reference				Reference	
Prediction	0	1		Prediction	0	1
0	True Negative	False Negative		0	10	-70
1	False Positive	True Positive		1	0	0

Choosing the cut-off
Expected value approach

- For a given confusion matrix I can find the associated expected value

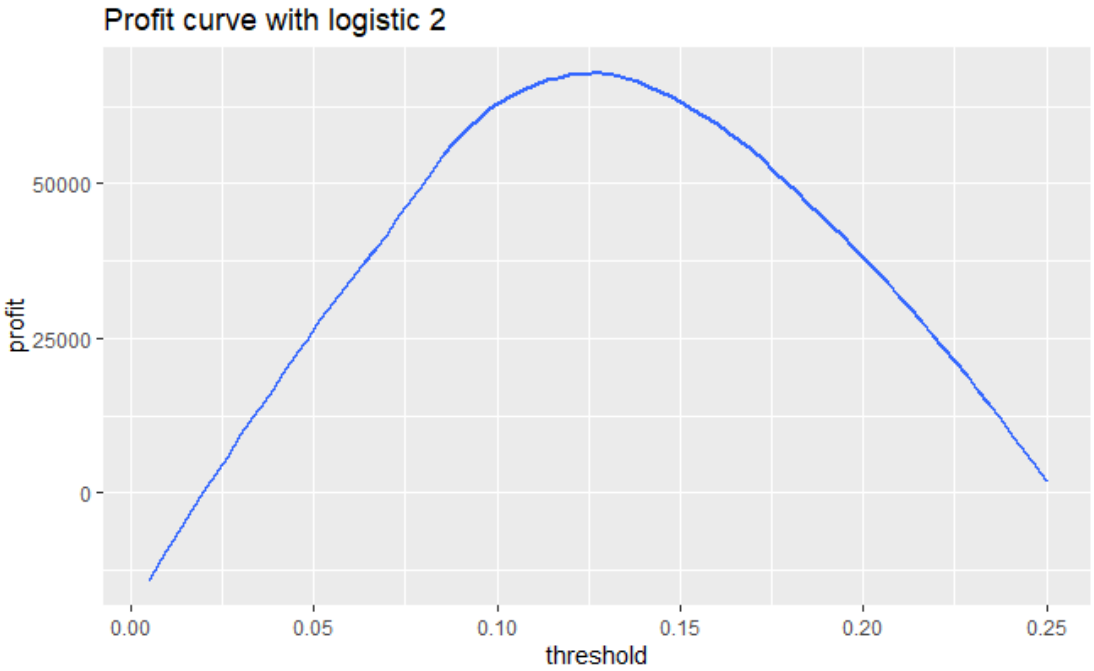
Confusion Matrix				Cost Matrix		
	Reference				Reference	
Prediction	0	1		Prediction	0	1
0	29445	4131		0	10	-70
1	2995	1298		1	0	0

Expected value = 29445 * 10 – 4131 * 70 = 5280

Or 18.95 cents per loan application

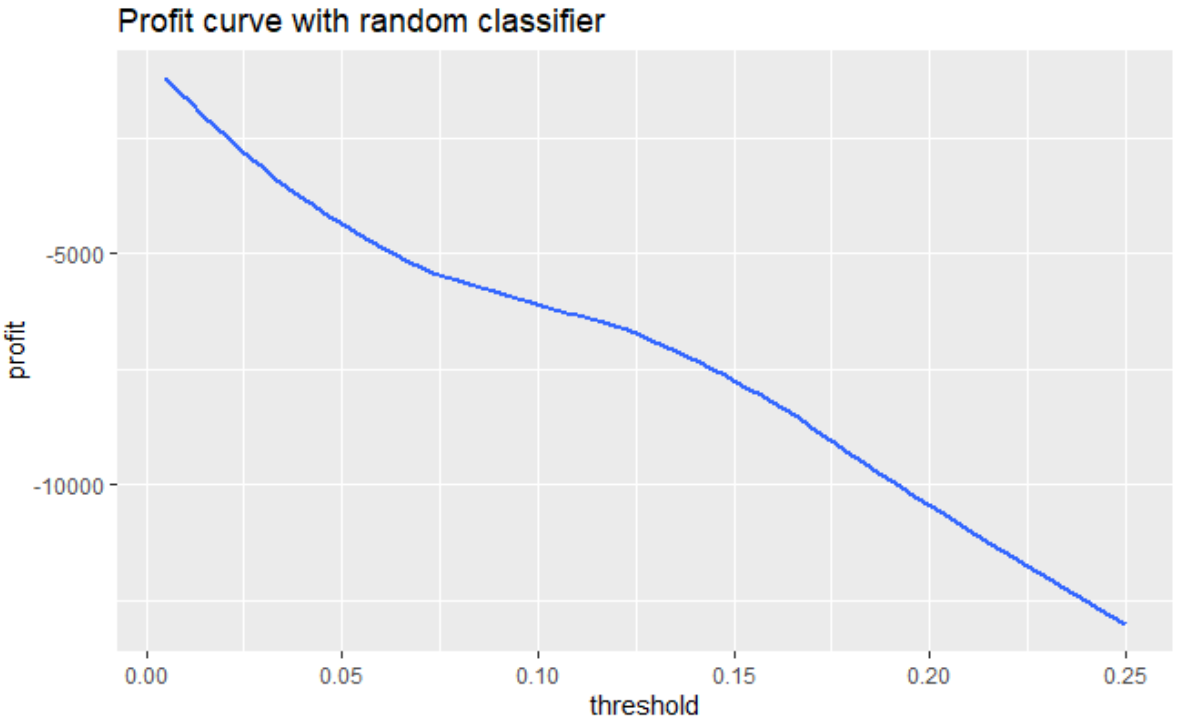
- But this is for one cut-off value and we have complete flexibility on what to choose
- How about we look at the profit as a function of the cut-off value (sensitivity analysis)

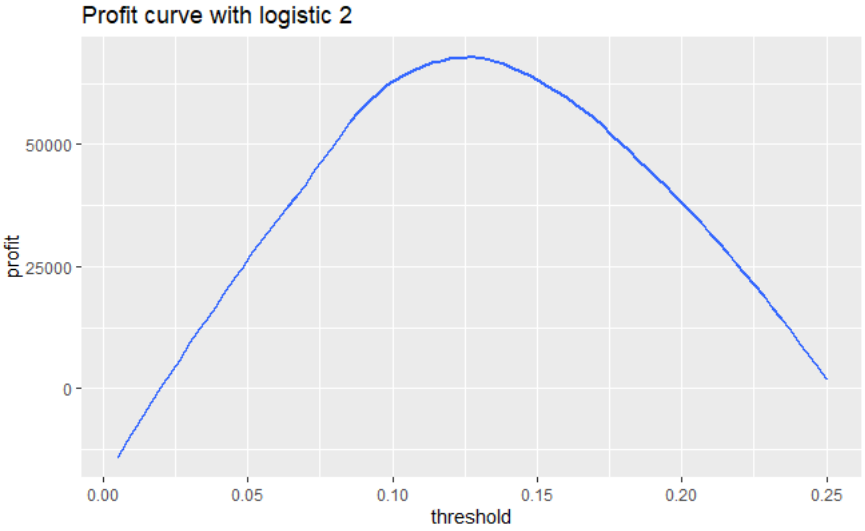
Choosing the cut-off Expected value approach

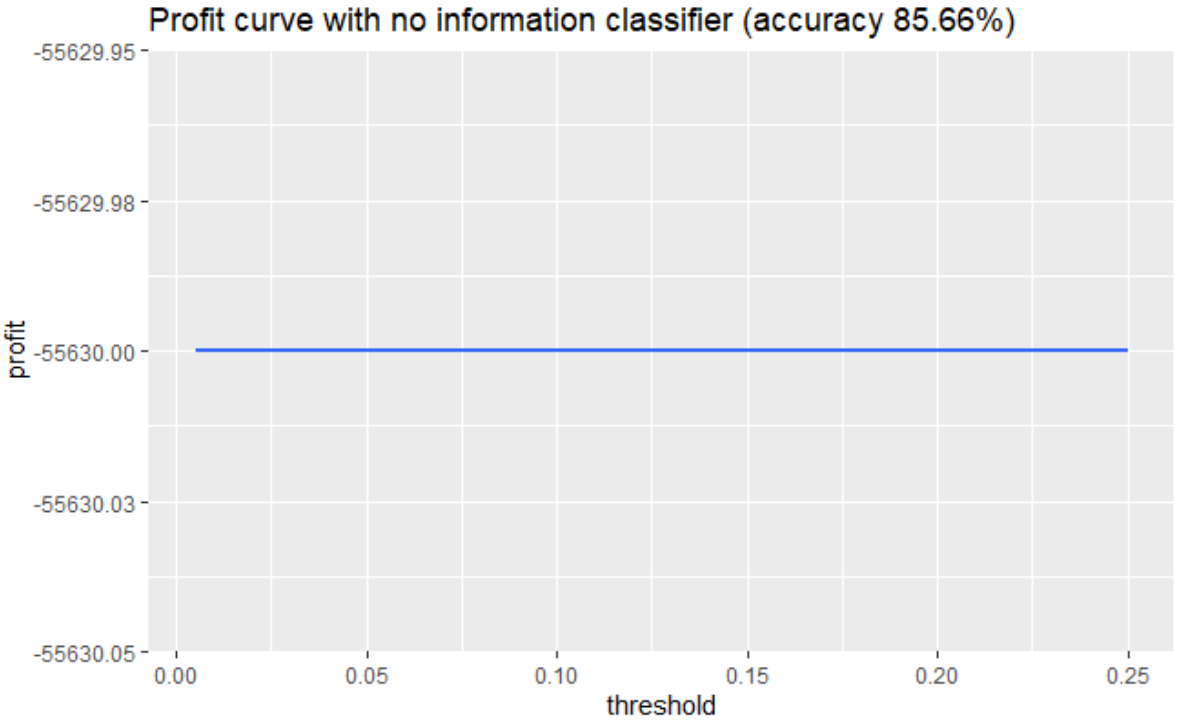


- Profit for different cut-off values is shown to the left
- In this case, a cut-off value of ~12.25% maximizes profits at just over $69,090 or $1.82 per loan application
- We can also use this curve to compare different models

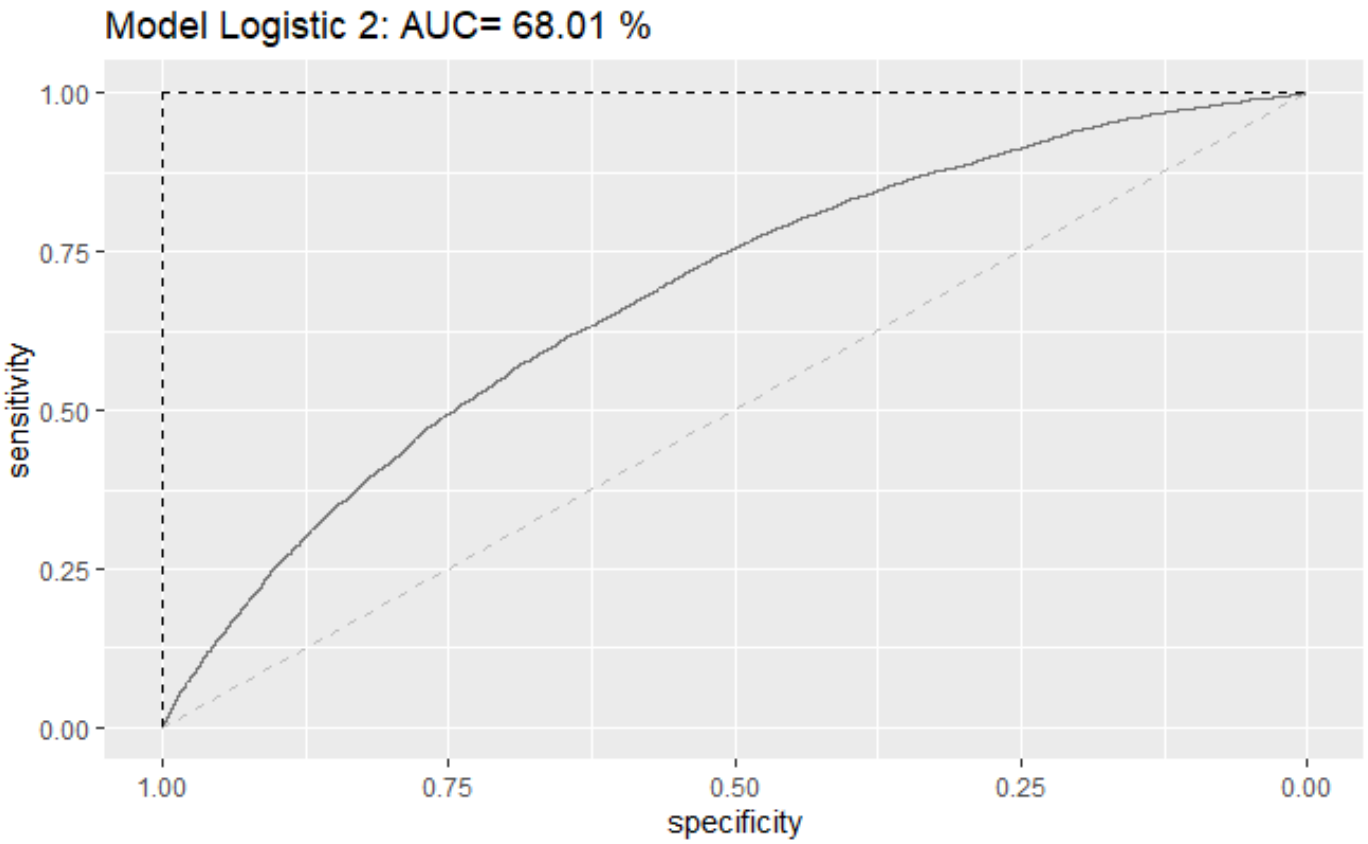
Comparing different models





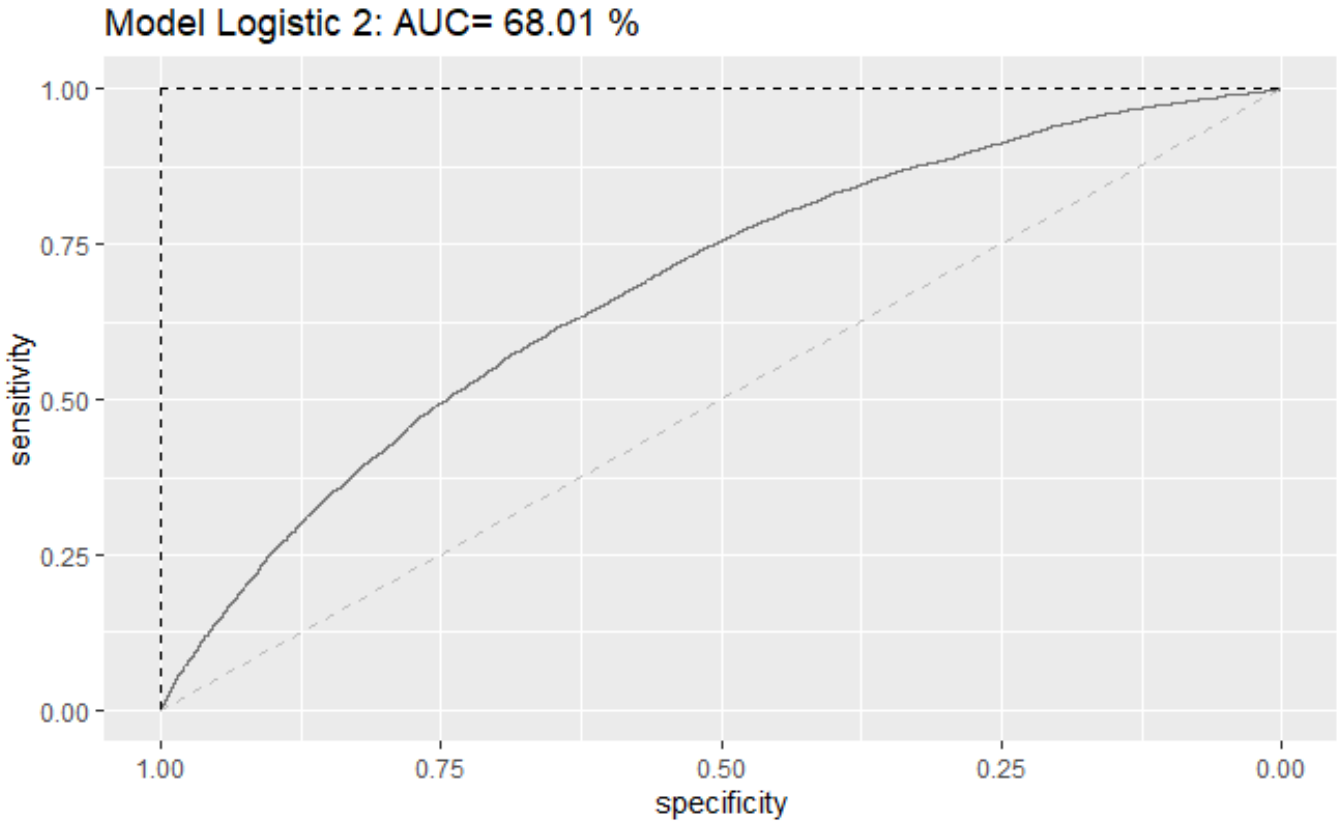


Assessing model performance



- The Receiver and Operating Characteristic (***ROC***) curve is a scatterplot of the model's sensitivity (True Positive rate) against the specificity (True Negative rate) for different cutoff values
- Typically the specificity axis (horizontal) is inverted
  - From 1->0 instead of 0->1
- The ROC curve shows for any given level of True Positives, what level of True Negatives we can expect
  - Obviously, the lower the better – so we want classifiers that are as close as possible to the top left corner (1,1)

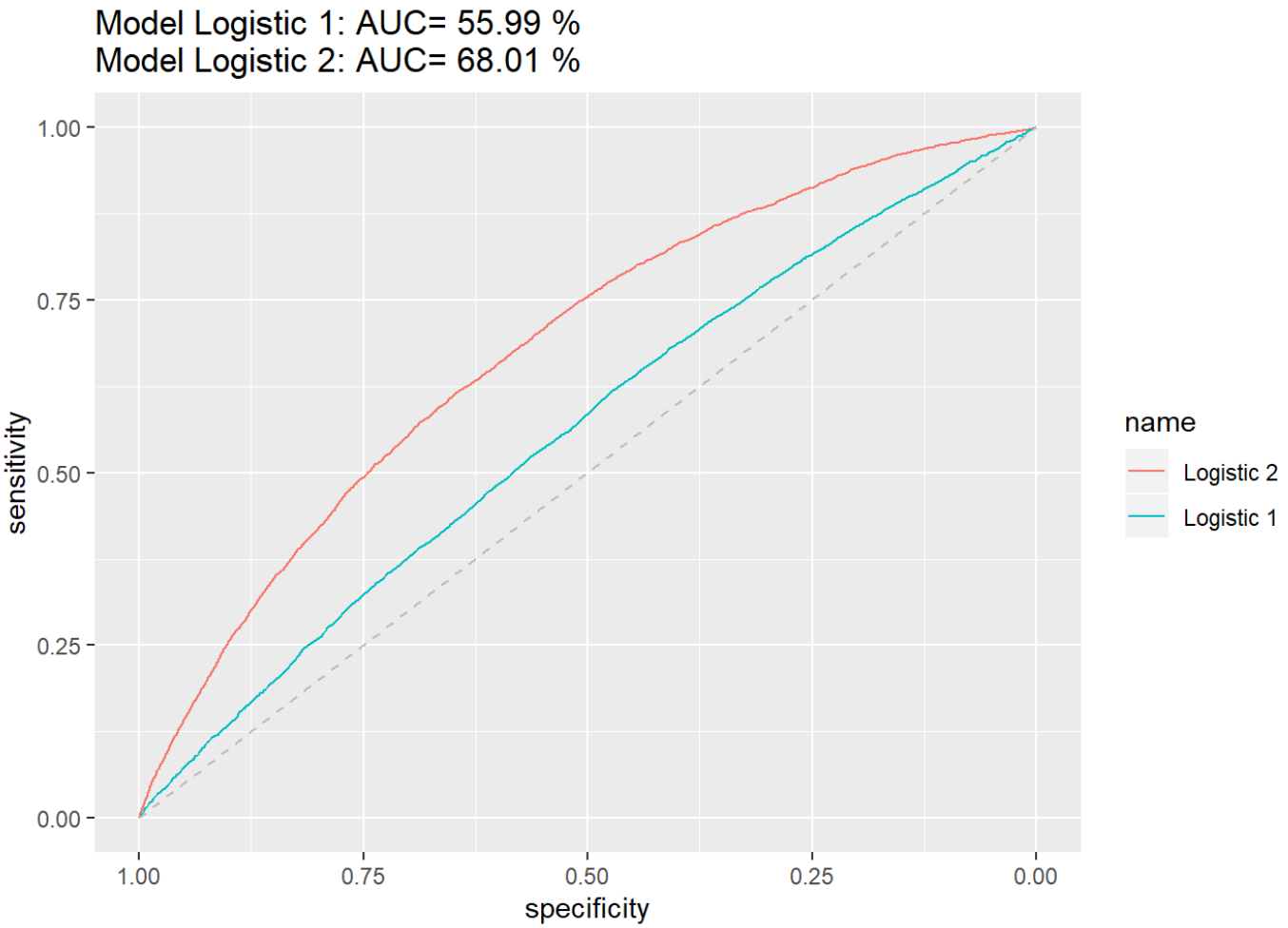
Assessing model performance



- We can use the area under the curve (AUC) to measure the predictive power of the model
  - AUC = 0.5 suggest that the model is no better than random chance (flipping a coin)
  - AUC=1 suggests that the model predicts perfectly (sensitivity=specificity=accuracy=100%)
  - The higher the AOC the better the model's predictive power

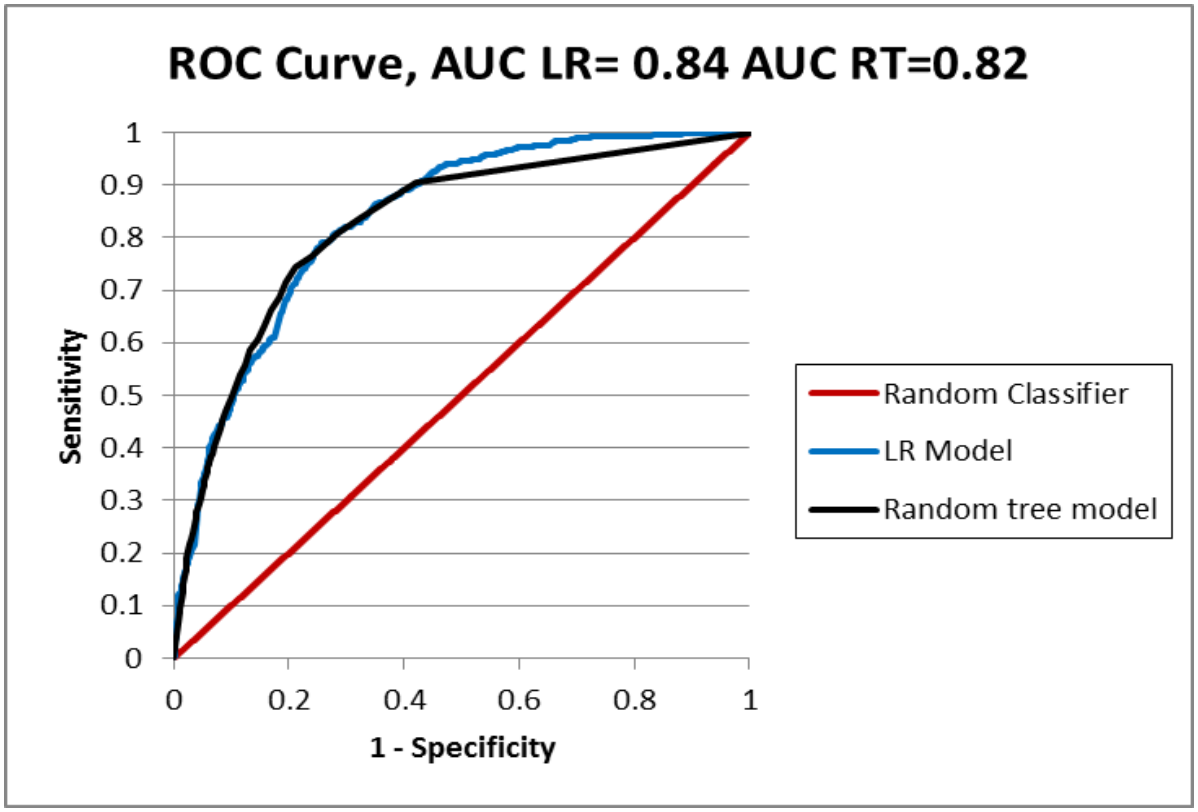
- ROC and AUC notions are not unique to logistic regression, they apply to any classification model

Comparing model performance



- We can use the ROC to compare models
- In this case the red model (logistic 2) ***dominates*** the blue model (logistic 1) – for any cut-off value the specificity is higher for the same sensitivity
  - This is not surprising as logistic 2 has more features and lower deviance (and AIC) than logistic 1
  - This results suggests that for any costs, the profit curve with logistic 2 will be higher than with logistic 1

Comparing model performance



- **How can I use ROC curves to compare these two models?**
  - In this case there is no strict dominance. The “blue model” preforms better at high sensitivity, the “black model” preforms better at high specificity
  - Then we choose the one that performs better in the region we want to operate at!

Out-of-sample testing

- So far everything was done in-sample
  - In-sample estimation of coefficients for probability model
  - In-sample estimation of cut-off point for prediction
  - In-sample estimation of errors, confusion matrix, profit, ROC and lift curves
- This is problematic as the sample may not be representative, way may be overfitting→ model may not work well on new data
- Best to do out-of-sample validation

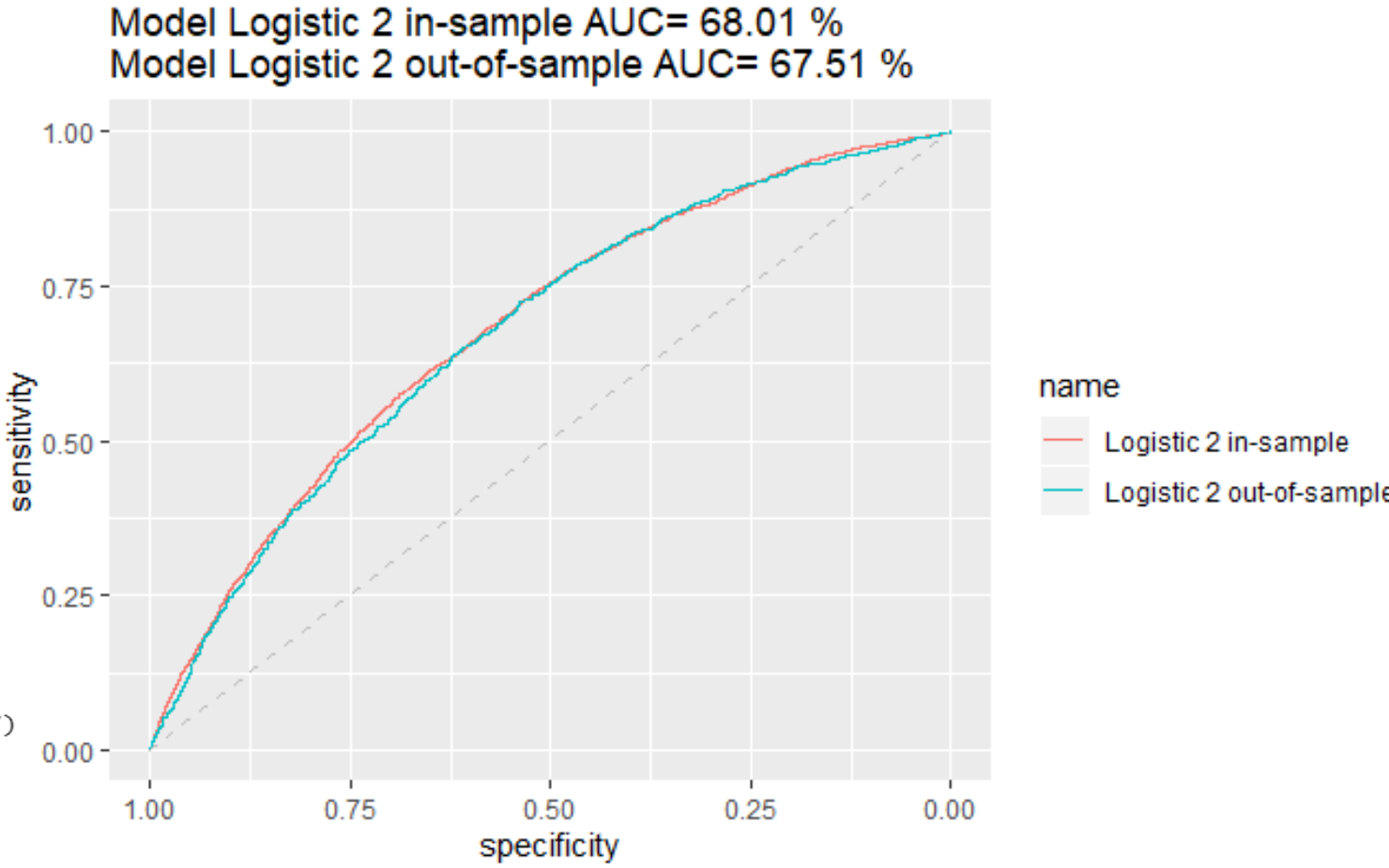
Out-of-sample validation

In sample (80 % of the data)

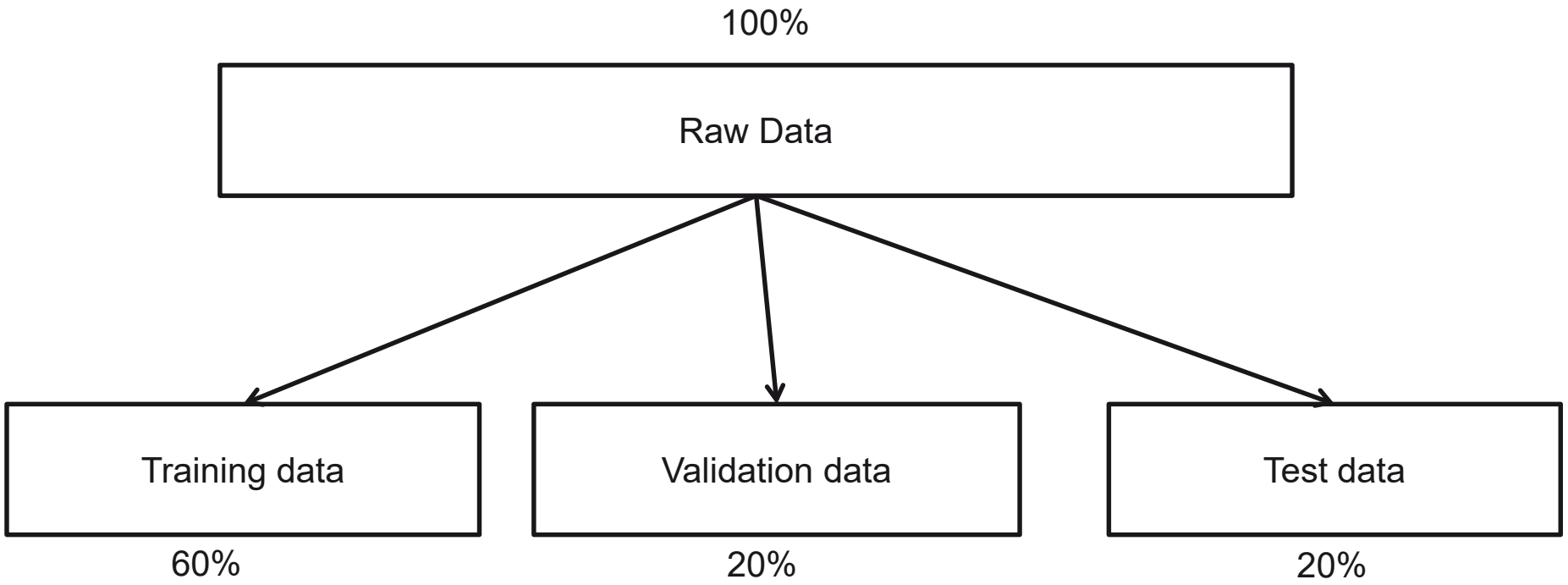
```
          Reference
Prediction     0     1
         0 23580  3308
         1  2371  1037

               Accuracy : 0.8125
                 95% CI : (0.8081, 0.8169)
    No Information Rate : 0.8566
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1618

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.23867
            Specificity : 0.90864
```

Out of sample (20% of the data)

```
          Reference
Prediction    0    1
         0 5865  823
         1  624  261

               Accuracy : 0.8089
                 95% CI : (0.7999, 0.8177)
    No Information Rate : 0.8569
    P-Value [Acc > NIR] : 1

                  Kappa : 0.1566

 Mcnemar's Test P-Value : 1.939e-07

            Sensitivity : 0.24077
            Specificity : 0.90384
```



Three-way data partitioning



- **In calculating the optimal cut off we need to partition data into three samples**
  - Training → estimate the model (typically larger than the other two)
  - Validation → estimate the cutoff (or any other hyper-parameter)
  - Test → estimate overall performance

Three-way data partitioning

- **Without splitting the data (see slide 34)**
  - Best threshold = 12.25% with a profit of $1.82 per loan
- **Using three-way partitioning method**
  - In sample: Model estimation
  - Validation set: Best threshold =11.75% with a profit of $1.81 per loan
    - Lower threshold → more conservative because now we make more errors
    - Lower profit because of the increase in error
  - Testing set: out-of-sample profit of $1.79 per loan
    - Lower than both the in-sample and the validation
- **This deterioration in performance is typical**
  - Deterioration could be worse for models that suffer from overfitting

k-fold Cross-validation

- Splitting the data in training and validation means that
  - we reduce the data used for training which may be a problem if we have a relatively small dataset
  - we don't get to use every data point for estimation and for validation (only one of the two)
- K-fold cross-validation overcomes this problem:
  - Randomly divide data into K equal-size groups (referred to as folds)
  - Use the first fold as validation and the other K-1 for training and compute Accuracy or AUC
  - Repeat this K times
  - Use the average Accuracy or AUC to estimate model performance
- Typically set K = 5 or 10

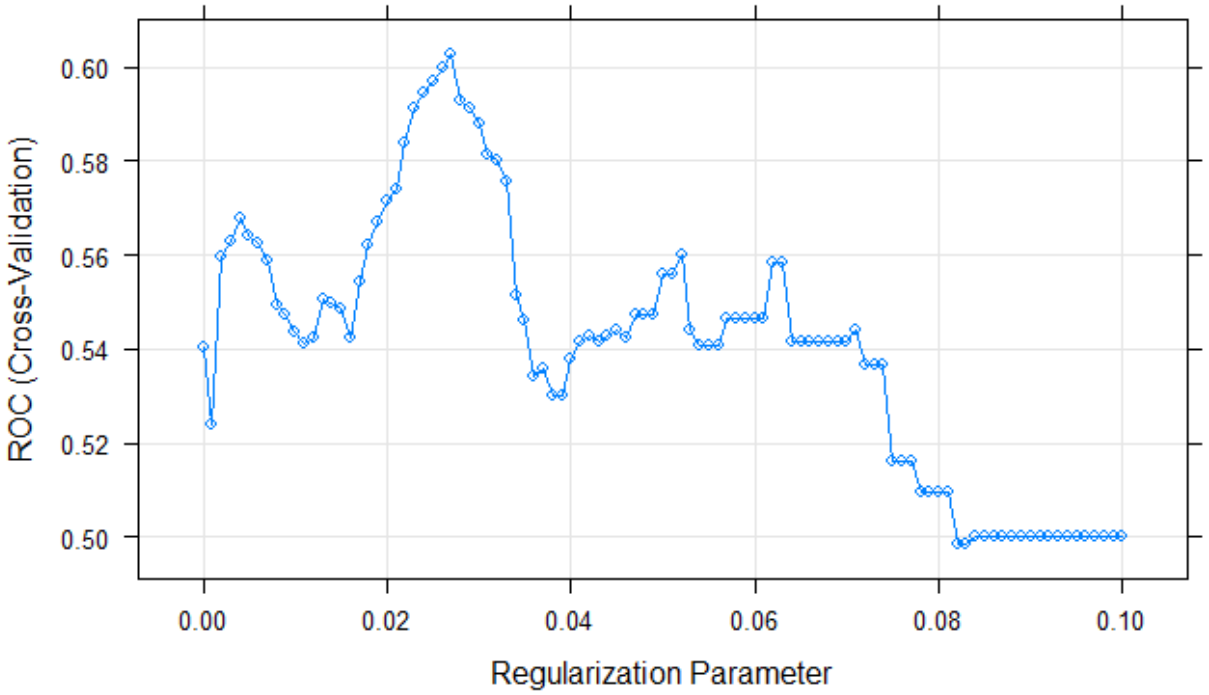
Data

Validation	Train	Train	Train	Train
1	2	3	4	5

Logistic Regression and Regularization

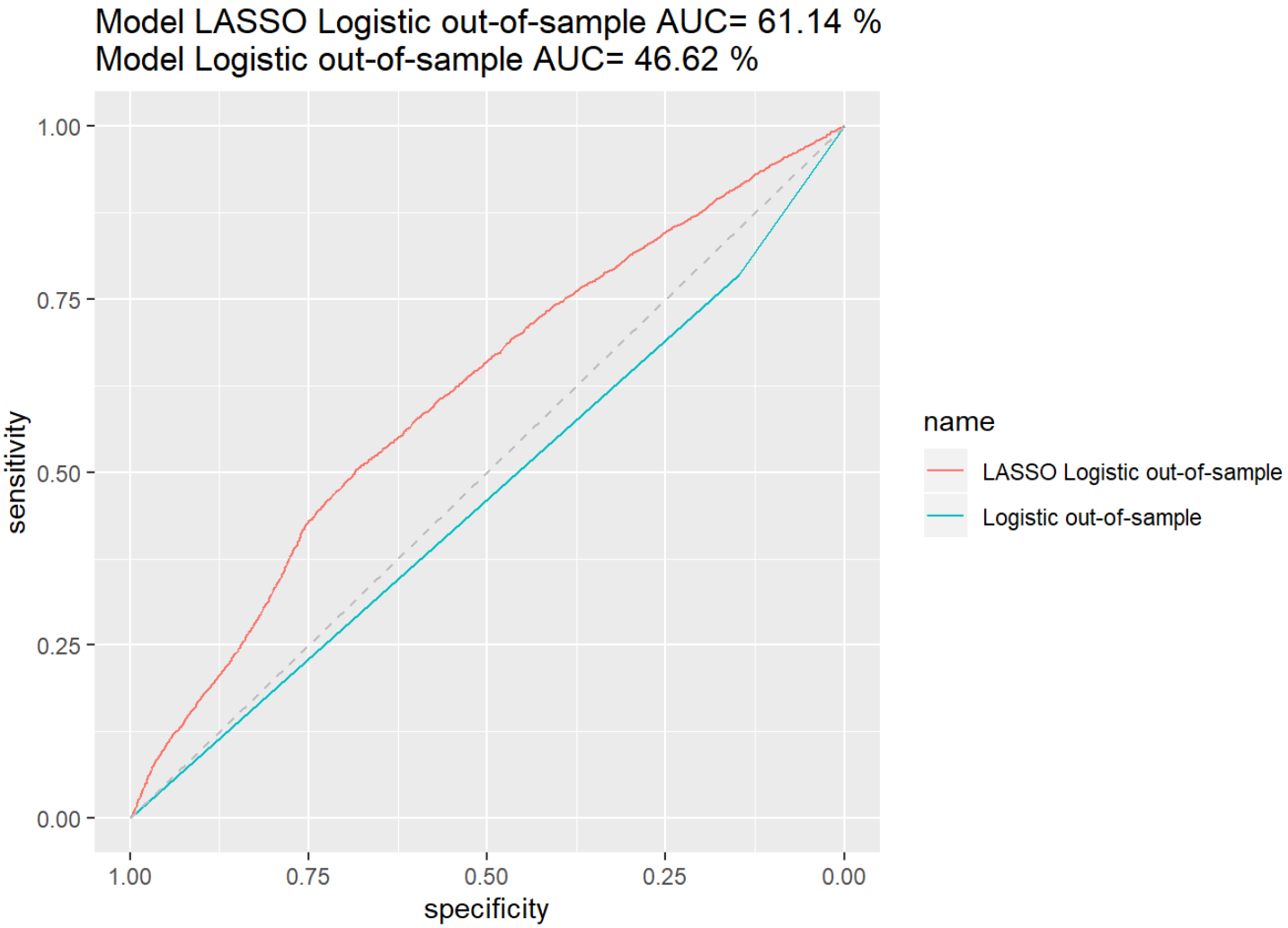
- We can also do logistic regression using the idea of regularization
  - Similar to linear regression
- Instead of maximizing log-Likelohood to estimate the model's coefficients we can maximize log-Likelihood minus λ times the sum of the absolute value of estimated coefficients
  - As in the OLS case, the penalty has the effect of shrinking the estimated coefficients towards zero
  - The penalty parameter λ is chosen to maximize performance out-of-sample (typically through k-fold cross validation)
  - Remember to standardize any features before running LASSO

LASSO performance



- Select 1% of the data for training
- Estimate a huge model (168 coefficients) using
  - Logistic Regression
  - LASSO logistic regression with 10fold cross validation
- The chart shows AUC (based on 10-fold CV) against the parameter λ
- Highest AUC at $\lambda = 0.027$
  - This is somewhat sensitive to training sample, but the performance is not so sensitive

Performance of LASSO



Select 1% of the data for training (<400 loans)

Estimate a huge model (168 coefficients) using

- Logistic Regression
- LASSO logistic regression with 10fold cross validation

Use both models to construct out-of-sample ROC curves (on the 99% testing data)

Logistic regression checklist

- Step 1. ICE the date, scatter plots & correlation tables
- Step 2. Develop model & Feature engineering. If in doubt start simple
- Step 3. Estimate the model using software (logistic / LASSO)
- Step 4. Post Estimation
  - Examine significance of individual variables: check p-values, drop out non-significant variables (e.g., p-value > 5%)
  - Check goodness of fit: deviance, ROC curve
  - Select cut-off value if needed (e.g., use profit curve)
  - Check for out of sample performance (ROC, precision, specificity, sensitivity, profit)
- Step 5: Repeat steps 2-4 to assess and compare competing models. Choose the simplest model that has good enough explanatory power and makes intuitive sense

Rules of thumb

- The number of observations should be at least $10 \times \frac{m+2}{min(p,1-p)}$ where m is the number of explanatory variables and p is the probability of a failure
  - This is higher than the rule of thumb for linear regression

- We can construct confidence intervals for the coefficients of estimation and for odds ratios. Don't use variables that are not significant (say at the 5% level) in your model due to the problem of overfitting

- Don't rely on R-square measures for classification problems (they will tend to be low). Instead check precision, sensitivity, and specificity, ROC curves and AUC.

Multinomial Logistic Model

- The model extends to classification in more than two categories
- Say there are three categories $Y = \{R, G, B\}$
- Choose one category as the base, say R
- Define the two risk factors
  - $U_R = 0$, $U_B = \beta_0 + \beta_1 X_1 + \cdots \beta_m X_m$, $U_G = \gamma_0 + \gamma_1 X_1 + \cdots + \gamma_m X_m$
  - The probabilities that Y takes any of the three categories given X are
  - $Prob(Y = B|X) = \frac{\exp(u_B)}{1+\exp(U_B)+\exp(U_G)}$
  - $Prob(Y = G|X) = \frac{\exp(u_G)}{1+\exp(U_B)+\exp(U_G)}$
  - $Prob(Y = R|X) = \frac{1}{1+\exp(U_B)+\exp(U_G)}$
- We can estimate the model using the maximum likelihood principle
  - Confusion matrix and ROC curve are now multidimensional

Oversampling

- It is often the case that we are interested in rare events (<1% of the data is “success”) such as fraudulent credit card transactions
- Therefore, sampling a random subset (eg training set) may yield too few events, making it difficult to train a model
- To give the model a better “chance” to train, we may want to oversample “successes” so that the dataset is more balanced
- Oversampling does not affect the estimated coefficients (except the intercept), p-values, or the ROC
- Predicted probabilities using oversampling will need to be adjusted
- Oversampling approach
  - Oversample in the training set but not in the validation set

Next lecture

Course contents (first part of the course – Kamalini)

- Session 1: The Art & Science of Regression Models For Prediction
- Session 2: More on Using Linear Regression For Prediction
- Session 3: Workshop I – Engineer an algorithm that sets interest rates for new Lending Club loans
  - Group assignment 1, due 6 days after the workshop
- Session 4: Classification using Logistic Regression
- **Session 5: Workshop – Invest in a portfolio of Lending Club loans**
  - Individual project 1, due 13 days after the end of the workshop

Course contents (second part of the course – Kanishka)

- See canvas syllabus

Innovation transforms lending

Lending Club is the world's largest marketplace connecting borrowers and investors, where consumers and small business owners lower the cost of their credit and enjoy a better experience than traditional bank lending, and investors earn attractive risk-adjusted returns.[4]

Here's how it works:

- **Customers interested in a loan complete a simple application at LendingClub.com**
- **We leverage online data and technology to quickly assess risk, determine a credit rating and assign appropriate interest rates. Qualified applicants receive offers in just minutes and can evaluate loan options with no impact to their credit score**
- **Investors ranging from individuals to institutions select loans in which to invest and can earn monthly returns**

The entire process is online, using technology to lower the cost of credit and pass the savings back in the form of lower rates for borrowers and solid returns for investors.