Session 3

Oleg Deev & Štefan Lyócsa

Masaryk University

FINTECH RISK MANAGEMENT
www.fintech-ho2020.eu

Assume that you are taking a 10 question test. Each question has 5 possible answers, you have to choose only **one**. Only one is correct. You do not know the answer to any of those questions, and you actually do not know anything about the topic of the test (say quantum field theory). How probable it is to have **at least** 6 correct answers (and to pass the test)?

Make a guess...

- Probability of a correct answer is $p = 0.20$.
- Answering questions is a Bernoulli trial, 1 if correct, 0 if not correct.
- You have $n = 10$ trials.
- We are interested in $x > 6$ correct answers.

The probability to have at least 6 correct answers can be solved using the **Binomial** probability distribution formula:

$$P(X > x) = \sum_{x=6}^{10} \binom{n}{x} p^{x} (1-p)^{(n-x)} = 0.00637$$

If you **learn** and increase the probability to $p = 0.5$, the overall probability of success increases to 0.37695.

We would like to model **success** (1) and **failure** (0) variable.

- What determines loan defaults?
- What influences consumers to buy a product?

Define a variable DEF_i to be 1 if the interest earned is negative, **partially** or **fully** defaulted loan, and 0 otherwise.

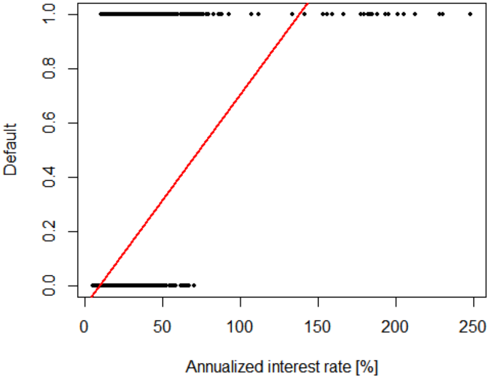
- Is it true, that the higher the annualized interest rate on the loan, the higher the probability that it defaults?

Why not to use simple linear regression? Let's see...

$$DEF_i = \beta_0 + \beta_1 Int_i + u_i$$

$DEF_i = \beta_0 + \beta_1 Int_i + u_i$ The estimated coefficients are:

$$DEF_i = -0.074 + 0.0078 \times Int_i + \hat{u}_i$$



Instead of modeling probability p_i, we could model the **odds**.

$ODDS_i = \frac{p_i}{(1-p_i)}$

This solves the ceiling, as $ODDS_i > 0$, but the lower bound is at 0. We could take the log (natural logarithm):

$log(ODDS_i) = log(\frac{p_i}{(1-p_i)}) = logit(p_i)$

$logit(p_i) = \beta_0 + \beta_1 \times Int_i$

We could get back to the probability as:

$p_i = \frac{e^{\beta_0 + \beta_1 \times Int_i}}{1+e^{\beta_0 + \beta_1 \times Int_i}}$

Coefficients are estimated using **maximum likelihood**. Each loan is a Bernoulli trial, with 1 if the loan defaulted and 0 if not. The probability of observing a default can be modeled as:

$$\underset{\hat{\beta_0},\hat{\beta_1}}{min} \rightarrow LL(\beta_0, \beta_1) = \sum_{i=1}^{n} y_i log(p_i) + (1-y_i) log(1-p_i)$$

Estimating the model leads to the following coefficients:

$$log(\frac{p_i}{1-p_i}) = -4.35 + 0.085 \times Int_i$$

A **unit increase** in annualized interest rate, increases the **log odds** by 0.085. Perhaps, it could be easier to transform the result to probabilities...

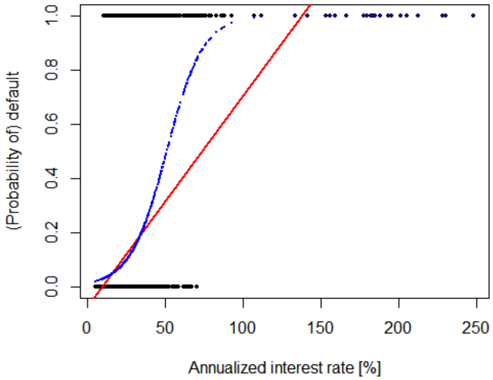
$$log(\frac{p_i}{1-p_i}) = -4.35 + 0.085 \times Int_i$$

A **unit increase** in annualized interest rate, increases the **log odds** by 0.085. Perhaps, it could be easier to transform the result to probabilities.

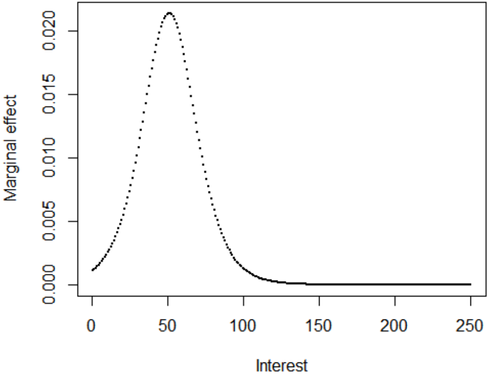
$$p_i = \frac{e^{-4.35+0.085 \times Int_i}}{1+e^{-4.35+0.085 \times Int_i}}$$

Obviously, the effect of Interest rate depends on the level of Interest rate. It is non-linear.

$$p_i = \frac{e^{-4.35+0.085\times Int_i}}{1+e^{-4.35+0.085\times Int_i}}$$



How the effect of interest rate changes with the **level** of the interest rate?



Using the data on micro-loans from the P2P market, we would like to:

- estimate a logit model of default using a test sample.
- predict the probability of a default.
- evaluate the prediction.

Estimation

We will use the same sample splits as in previous Case studies. We first need to define the DEF_i variables, for test and validation samples.

- `tst$DEF = (tst$RR2<0)*1`
- `val$DEF = (val$RR2<0)*1`

Now we estimate the model:

- `m12 = glm(formula = DEF ~ new+ver3+ver4+...+nopearlyre`

```
family = binomial(link = "logit"),data = tst)
summary(m12)
```

Prediction

Now forecasting exercise. We extract log odds.

- `ypred = predict(m12,new=val)`

We convert log odds to probabilities:

- `ppred=exp(ypred)/(1+exp(ypred))`

We create a table, where we compare the 'truth' with 'predictions':

- `true = val$DEF`
- `predicted = as.numeric((ppred>0.5)*1)`
- `TBL = table(predicted,true)`
- `TBL`

Evaluation

- What is the **accuracy** of our predictions?

- `sum(diag(TBL))/100`

- What is the **sensitivity** of our predictions?

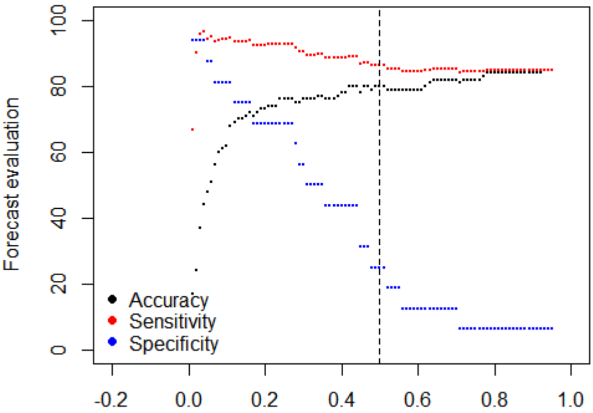
- `TBL[1,1]/sum(TBL[1,])`

- What is the **specificity** of our predictions?

- `TBL[2,2]/sum(TBL[,2])`

We have **high** overall accuracy, but **small** specificity. Note that we classify a loan to be defaulted if $p < 0.5$. What if 0.5 is not correct?

Let's take a look how our evaluation changes, if we change our **threshold** variable.



Session 3

Oleg Deev & Štefan Lyócsa

Masaryk University