To fail or not to fail?

Startup success prediction based on historical data

APPS UCU Econometrics Final Project, Spring 2019

Sofiya Hevorhyan
Iryna Popovych

INTRO

When choosing a topic for this project, we wanted to find something that would be interesting for us to do and valuable in terms of business perspective.

Tired of looking at dozens of datasets that were not very informative, and inspired by recent speech from our guest speaker Oleksandr Komarevych, we decided to choose startups as our topic.

BUSINESS OBJECTIVE

Investment strategies for start-up companies are usually based just on intuition or past experience. The question we pose here is,

Can we perform some analysis that can be used to identify relevant factors and score prospective startups based on their potential to be successful?

DATA

We have a dataset from CrowdAnalytiX that represents startups and covers information about various aspects of company, cofounders, investments, industry, activities of company, details about employees and technologies used.

Here is the link to dataset with information about 472 startups each having 116 characteristics.

You can take a look at data dictionary for descriptions of variables.

OUR PLAN

Data Preparation

- Data cleaning
- Missing value treatment
- Outliers treatment

Data Exploration

- Graphical exploration
- Hypothesis testing
- Principal component analysis

Feature Engineering

- Feature creation
- Determining feature importance
- Variable selection

Building a model

- Building logistic regression models
- Interpretation
- Comparing different results

Data transformation

Missing value treatment

Outliers treatment

1. Data Preparation

Data transformation

Missing value treatment

Outliers treatment

- We transformed strings to numerical, date, or factors where possible.

For example, the column Average.Years.of.experience.for.founder.and.co.founder transforms from to

High	3
Low	1
Medium	2
High	3
…	…

- We separated numerical and text data to make it more comfortable.

startup_df

cnt_df

char_df

Data transformation

Missing value treatment

Outliers treatment

- We defined percentage of missing values for every column.

```
mis_val<-sapply(startup, function(x) sum(is.na(x)))
percent_mis<-as.data.frame(round((mis_val/nrow(startup))*100,1))
```

- We separated all the rows with more than 40% missing not to use them in modelling.

```
# keeping only variables with less than 40% missing
new_var<as.character(pcnt_mis_var$variable[
                          which(pcnt_mis_var$Percent.Missing<40)])
new_startup<-startup[new_var]
```

- By doing this, 3 variables (columns) with lots of missing values were gone. On top of that, we reduced number of incomplete cases (rows which have missing values) by 56.

```
sum(!complete.cases(startup)) - sum(!complete.cases(new_startup))
-----> 56
```

Data transformation

Missing value treatment

Outliers treatment

- We calculated 10, 20, ….. 100 percentiles for every variable (column). Then created a function that replaces outliers with NA's using interquartile range rule.

  Outliers here are defined as observations that fall below Q1 - 1.5IQR or above Q3 + 1.5IQR.

```
remove_outliers <- function(x, na.rm = TRUE, ...) {
        qnt <- quantile(x, probs=c(.25, .75), na.rm = na.rm, ...)
        H <- 1.5 * IQR(x, na.rm = na.rm)
        y <- x
        y[x < (qnt[1] - H)] <- NA
        y[x > (qnt[2] + H)] <- NA
        return(y)
    }
```

- We cleared outliers, and after that used k-Nearest Neighbor, which is based on a variation of the Gower Distance to fill missing values.

```
cnt_df <- kNN(cnt_df, imp_var = FALSE)
```

Graphical exploration

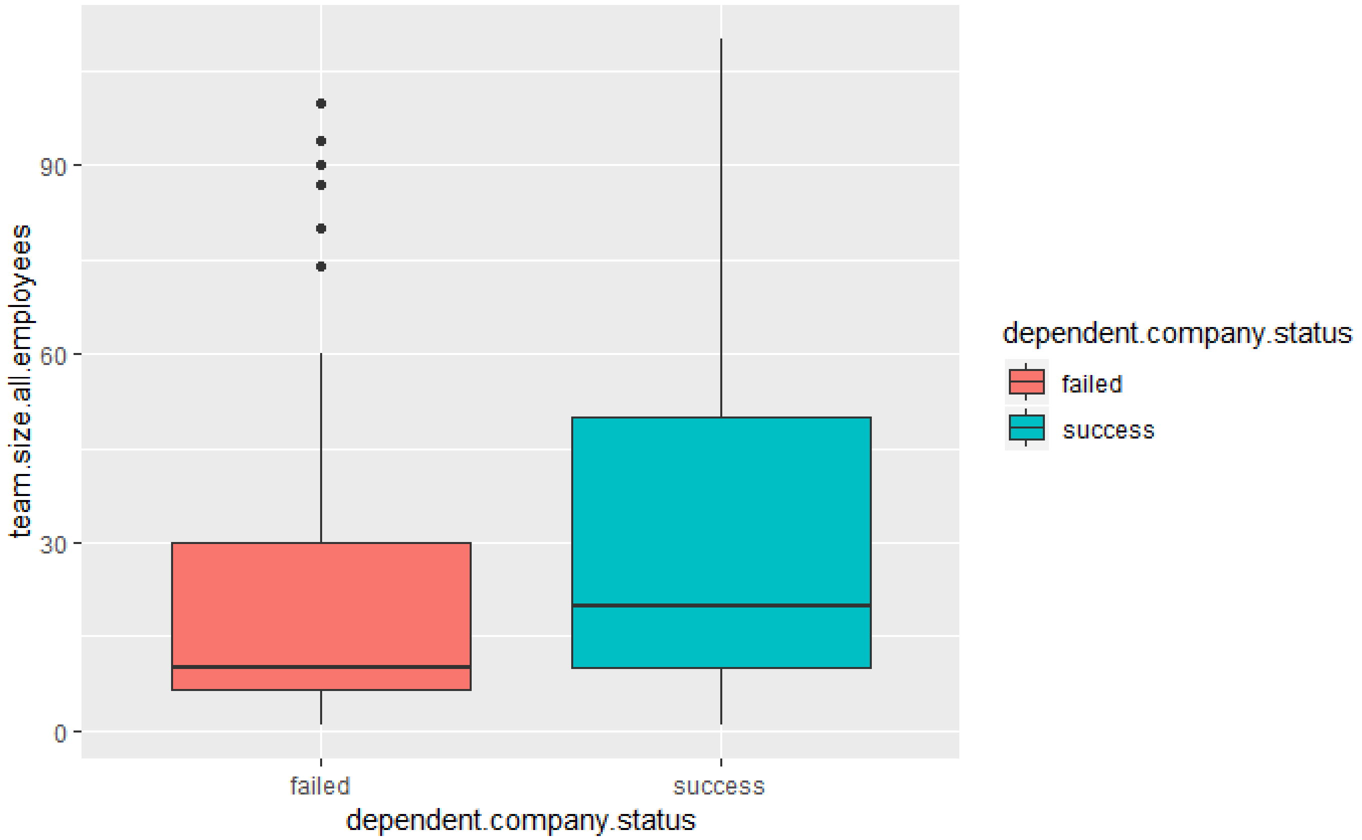
Hypothesis testing

PCA

2. Data Exploration

- We played with visuals to understand the data better. Here are some of them.

Company status by size of the team



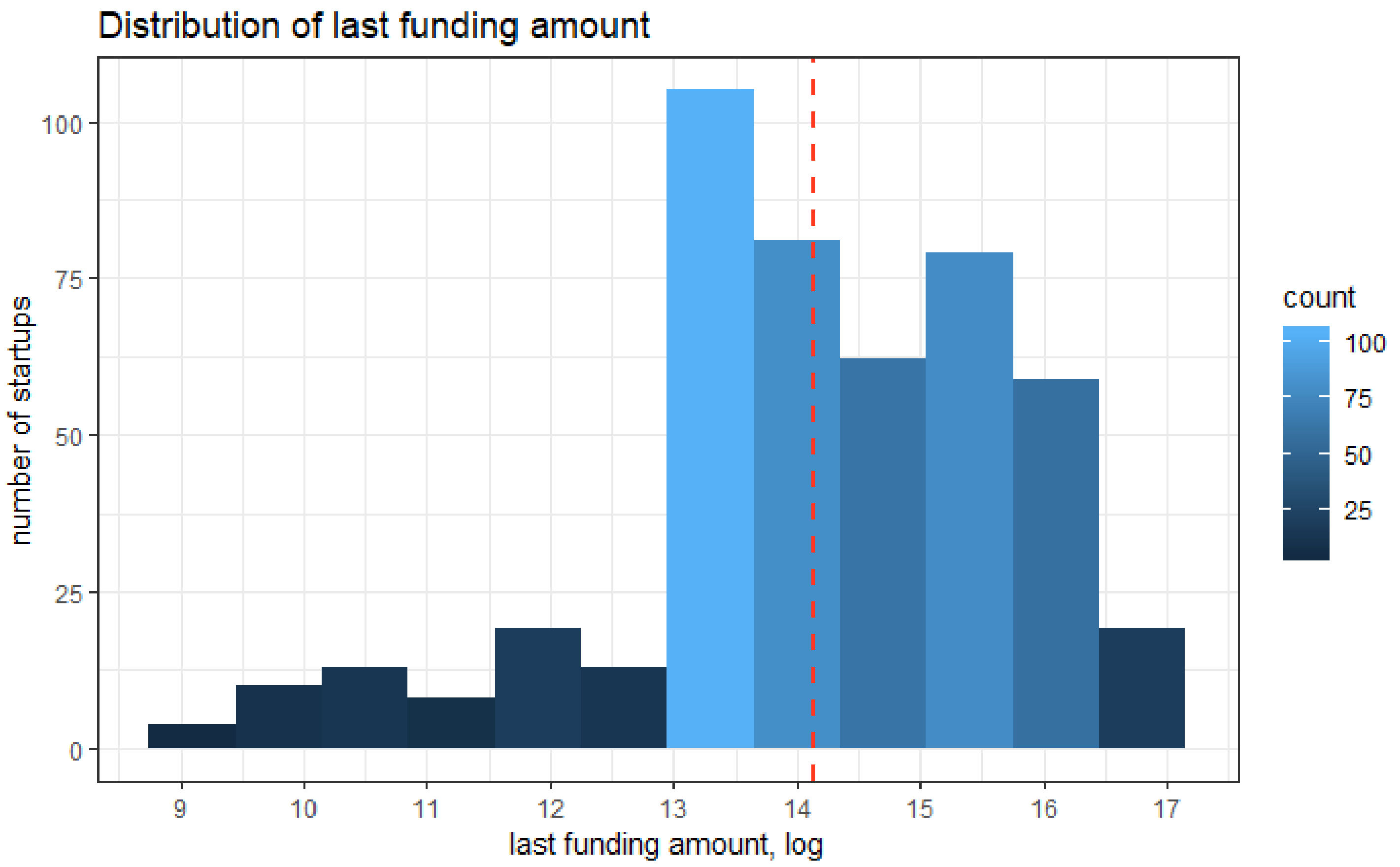
Graphical exploration

Hypothesis testing

PCA

- Graphical exploration
- Hypothesis testing
- PCA

- We played with visuals to understand the data better. Here are some of them:

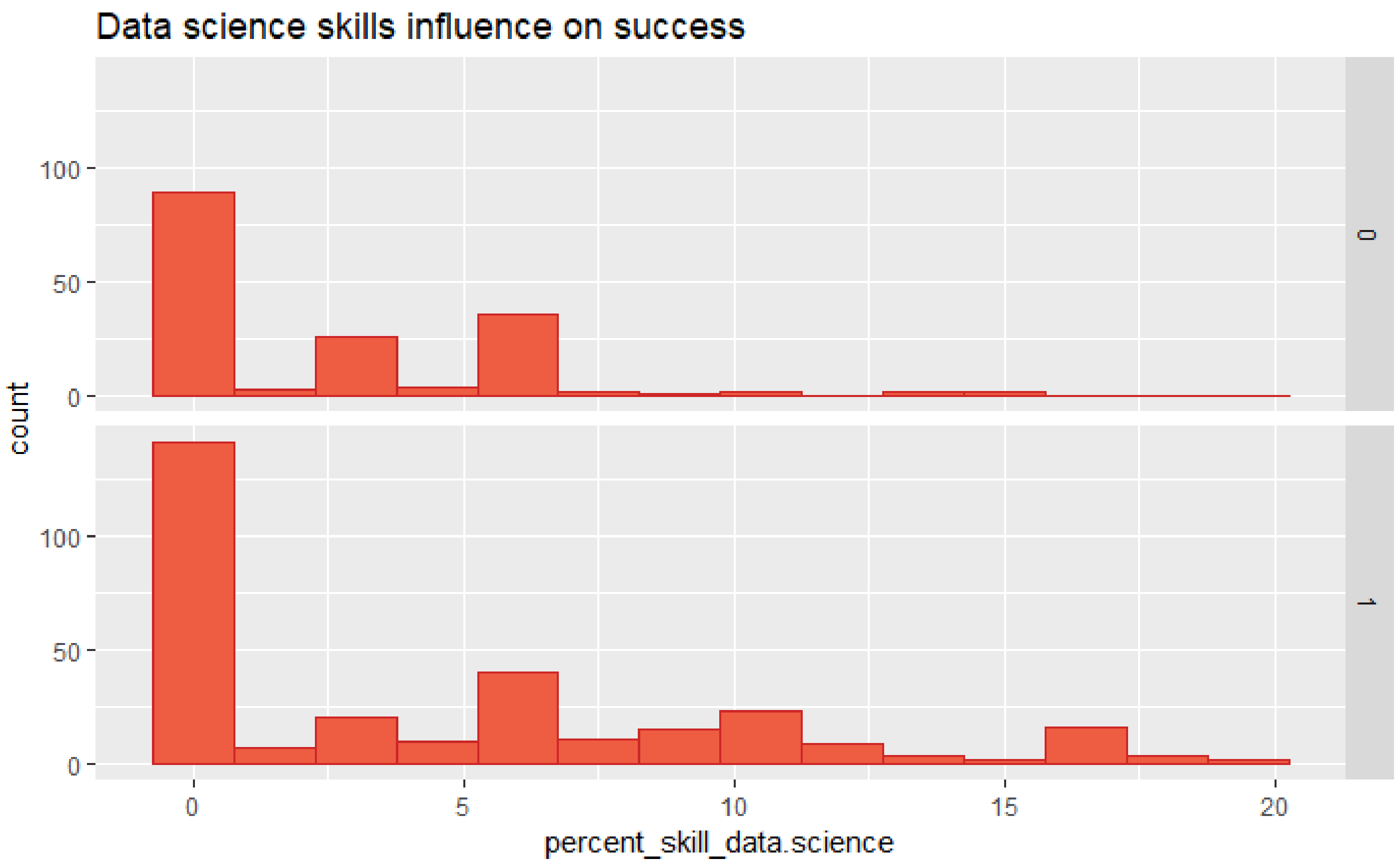


Graphical exploration

Hypothesis testing

PCA

- We played with visuals to understand the data better. Here are some of them:



Graphical exploration

Hypothesis testing

PCA

- We used 'T-test' for testing difference in mean of an independent variable to two categories of dependent:

Let's see if team size is an important feature or nor -- compare mean team sizes of companies who succeeded and who did not.

```
# t-test for checking difference in mean
t.test(team.size.all.employees~dependent.company.status, data=cnt_df)
        Welch Two Sample t-test

 data:  team.size.all.employees by dependent.company.status
 t = -3.4997, df = 341.26, p-value = 0.0005276
 alternative hypothesis: true difference in means is not equal to 0
 95 percent confidence interval:
  -11.29195  -3.16608
 sample estimates:
 mean in group 0 mean in group 1
        20.44311        27.67213
```

Here we can reject the null hypothesis of equality with a strong p-value of 0.0003, so, we can choose "Team.size.all.employees" for modeling.

- We also used "Chi-squared test" for testing interdependency for a categorical variable and dependent categorical. We used it to decide whether to use variable in modeling or not.

Graphical exploration

Hypothesis testing

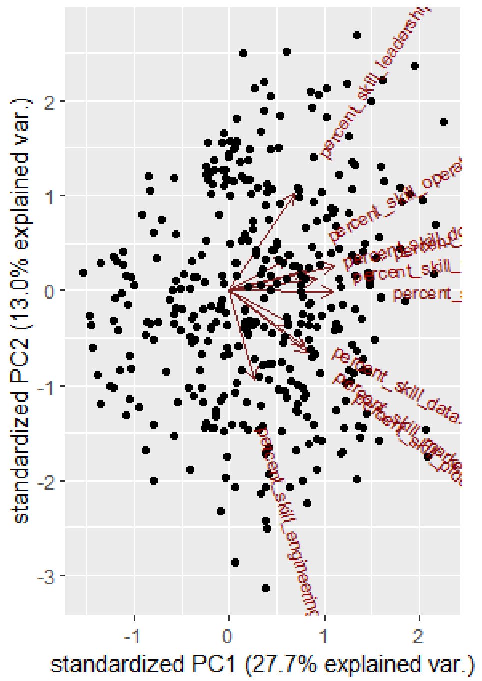
PCA

- We decided to perform Principal Component Analysis because our dataset is too wide.

  In such cases, where many variables are present, you cannot easily plot the data in its raw format, making it difficult to get a sense of the trends present within. PCA allows you to see the overall "shape" of the data, identifying which samples are similar to one another and which are very different.

- However, even though our dataset is wide, we do not have enough observations.

  After dozens of attempts, the best result we got was the PC which described less than 30% of variance, which is too little.



Feature creation

Determining importance

Variable selection

3. Feature Engineering

Feature creation

Determining importance

Variable selection

- We created additional features in given data to make it more meaningful which will help in analysis / modeling.

  - For example, variable “Investors” has list of investors for the company separated by ‘pipline’ symbol. We can create ‘Count.of.investors’ variable which will help in analysis.
  - We can also create some ratios to reduce number of variables. For example, we created lastfunding.amount vs. age of company ratio, as these two variables are dependent.

Feature creation

Determining importance

Variable selection

- We needed to select features for modeling, because we aren't able to use all of them.

- One of the ways to prove if the feature is meaningful and can be used in modelling is performing hypothesis testing for correlation, dependence etc., which we mentioned earlier in 'hypothesis testing' section.

- After analyzing different ways of determining importance, we decided to use random forest to select the most relevant features.

Advantages of random forest:

- They can deal with messy, real data. If there are lots of predictors, it has no problem.
- It automatically does a good job of finding interactions as well.
- There are no assumptions that the response has a linear (or even smooth) relationship with the predictors.

Feature creation

Determining importance

Variable selection

- One more approach that we sed for variable selection is Information Value.
    - Information Value (IV) for logistic regression is analogous to correlation for linear regression.
    - Information value tells us how well an independent variable is able to distinguish two categories of dependent variables.
    - We selected variables with IV of 0.1 to .7 for modeling.

```
# selecting variables with good information values
var<-IV[which(IV$InformationValue>0.1),]
var1<-var[which(var$InformationValue<0.7),]
final_var<-var1$Variable
```

Logistic regression

Comparing results

Interpretation

4. Building a model

Logistic regression

Interpretation

Comparing results

- Remember? We need to predict whether company will succeed or not:

```
startup <- read.csv(file="./data/CAX_Startup_Data.csv", header=TRUE,as.is=TRUE)
head(startup)
```

	Company_Name <chr>	Dependent.Company.Status <chr>	year.of.founding <chr>	Age.of.company.in.years <chr>
1	Company1	Success	No Info	No Info
2	Company2	Success	2011	3
3	Company3	Success	2011	3
4	Company4	Success	2009	5
5	Company5	Success	2010	4
6	Company6	Success	2010	4

6 rows | 1-5 of 116 columns

Dependent variable

Logistic regression

Comparing results

Interpretation

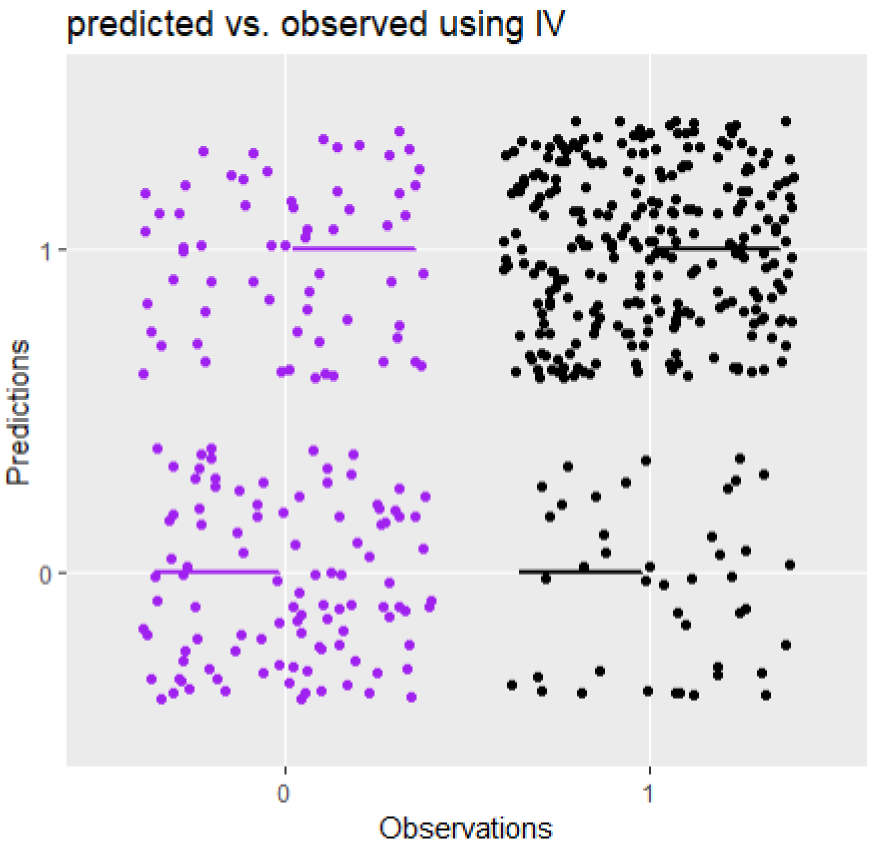
- We used different approaches to select predictors for the final model for our project.
    - randomForest + caret, varImp()
    - variable importance
- After that, we've built different models with these groups of features, including one mixed group, and compared results.
- Here, we present the model that appeared to be the best one. You can take a look at other models in the R-Notebook of the project with re-executable code.

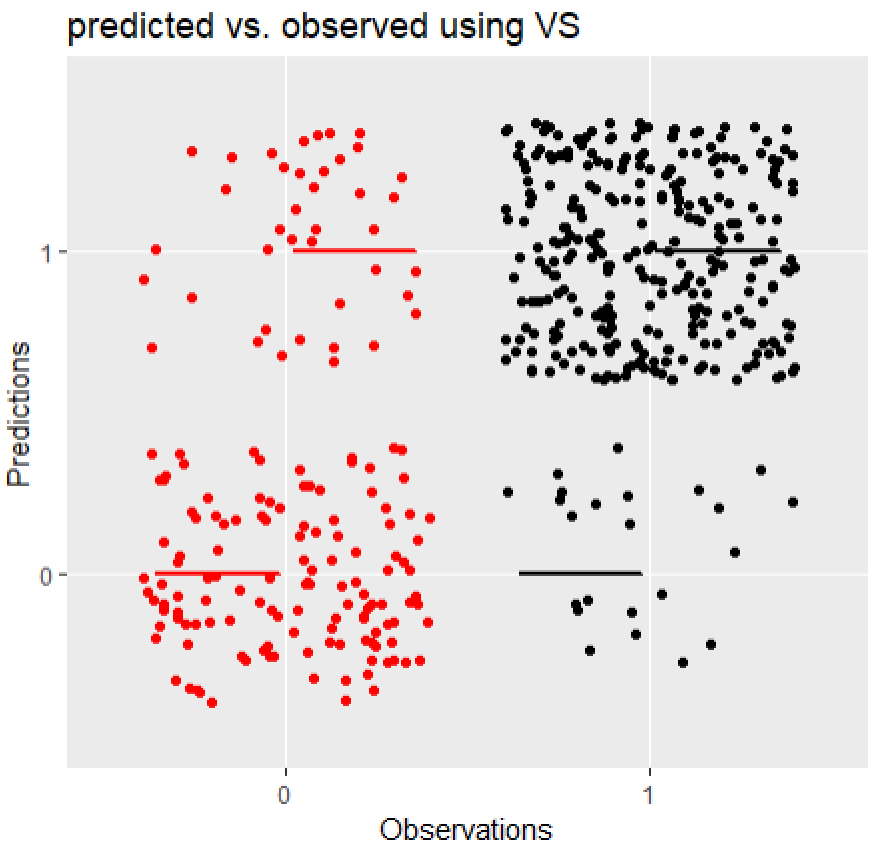
```
Coefficients:
                                       Estimate Std. Error z value Pr(>|z|)
(Intercept)                           -3.559649   1.207424  -2.948 0.003197 **
percent_skill_data.science             0.136885   0.031212   4.386 1.16e-05 ***
number.of.investors.in.seed            0.697207   0.151188   4.612 4.00e-06 ***
team.size.all.employees                0.010010   0.006220   1.609 0.107553
percent_skill_leadership              -0.092485   0.036339  -2.545 0.010925 *
percent_skill_engineering              0.021718   0.008258   2.630 0.008539 **
number.of.co.founders                  0.263153   0.118278   2.225 0.026090 *
experience.in.fortune.500.organizations 1.271004  0.339622   3.742 0.000182 ***
last.funding.amount                    0.200215   0.087595   2.286 0.022272 *
percent_skill_sales                    0.145185   0.042586   3.409 0.000651 ***
renown.score                          -0.143344   0.046112  -3.109 0.001880 **
percent_skill_operations              -0.147242   0.055933  -2.632 0.008476 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 613.39  on 471  degrees of freedom
Residual deviance: 435.52  on 460  degrees of freedom
AIC: 459.52
```

Logistic regression

Comparing results

Interpretation






```
          Reference
Prediction   0   1
         0 103  42
         1  64 263

               Accuracy : 0.7754
                 95% CI : (0.7351, 0.8123)
    No Information Rate : 0.6462
    P-Value [Acc > NIR] : 8.087e-10
```

```
          Reference
Prediction   0   1
         0 127  23
         1  40 282

               Accuracy : 0.8665
                 95% CI : (0.8325, 0.8959)
    No Information Rate : 0.6462
    P-Value [Acc > NIR] : < 2e-16
```

Comparison of Confusion Matrix for models with Information Variable methods and Variables Selection method

CONCLUSIONS

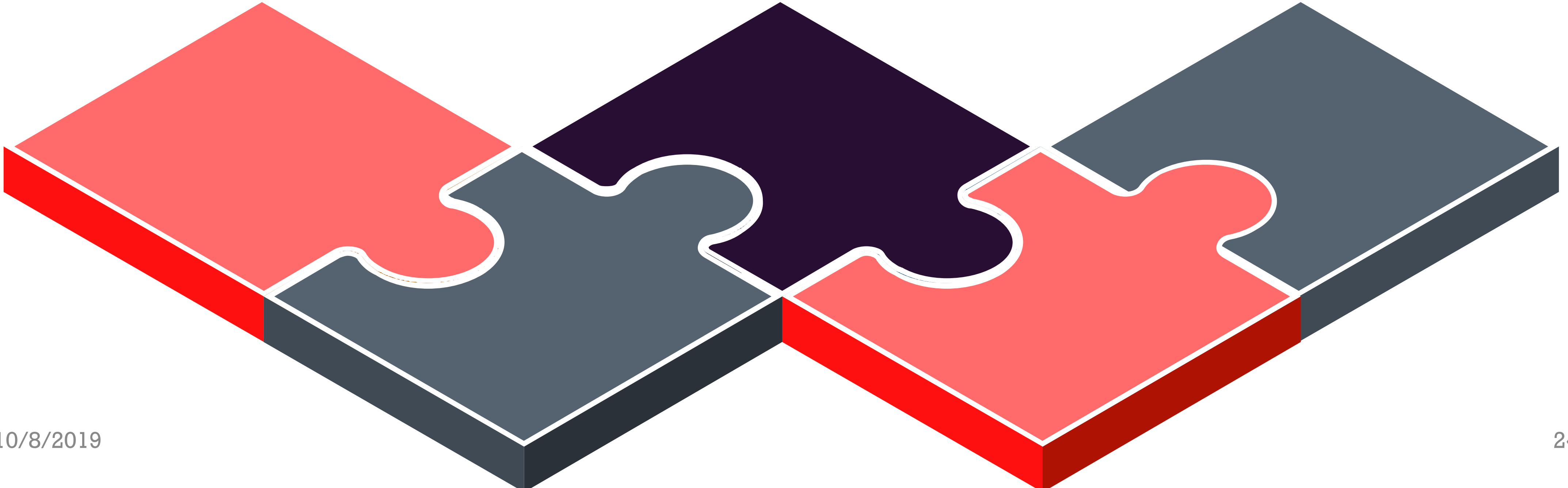
A Data cleaning and preparation is a hard work. It takes time.

B Now understand hypothesis testing to the very end, and know practical applications for it.

C We learned some new techniques for working with wide datasets.

D Feature selection was the most important part of our work. We used different methods, and tried to compare them to choose best.

E We have a piece of text data which we left untouched. We plan to work with text and multi-stage factors in the future.



AUTHORS

Iryna Popovych

Sofiya Hevorhyan

LINKS

/DataAnalysisR

link to dataset

THANK YOU