QUALITICS - BUTS

Patika.Dev - 110 VitrA Data Science Bootcamp Project

Group 2

Members: Regaip KURT, Canberk BULUT, Ayşe FIRAT, İsmail Sadi CESUR

Mentors: Utku BEZEK, Miray DOĞAN

Lecturer: Çağlar SUBAŞI

September, 2021

PURPOSE & Expected Value Framework

★ **Purpose:** Goal of this project is to reveal the features of manufacturing that leads to poor quality.

Various sensor datum are examined with different models to predict defective products.

★ **Expected Value Framework:** The expected benefit of the project is to find causes of the defective products and hence by designing the production line again to reduce the number of defective products.

Hence benefit can be financially described as the difference between the cost of defective products before the project and after the project.

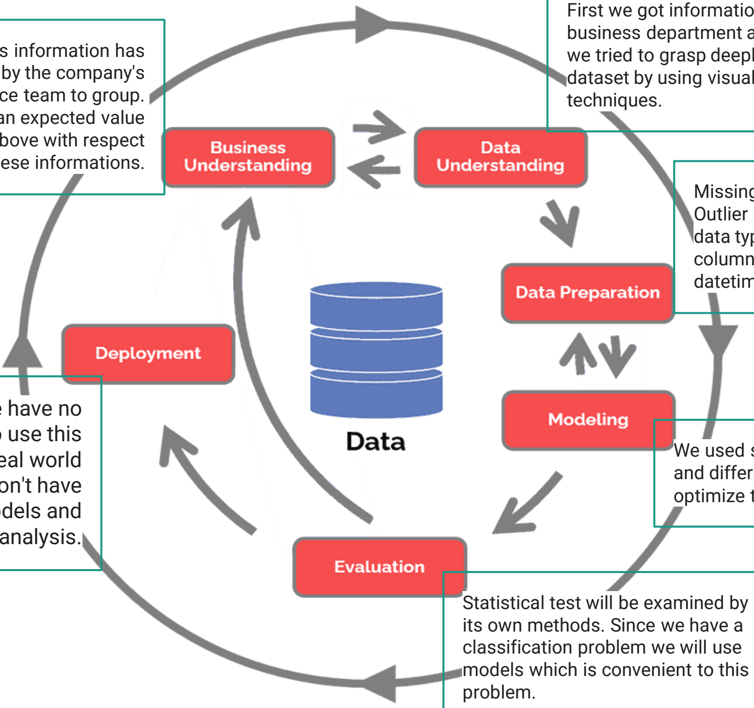
EV represents Expected Value and Expected Value equals to difference between Expected Benefit minus Expected Loss

$$EV = EB - EL$$

$$EV = \frac{n_1 + n_2}{2}\left(\frac{\sum_{i}^{n_1} c_b}{n_1} - \frac{\sum_{i}^{n_2} c_a}{n_2}\right) - f(c) - \mathcal{P}(f_p)$$

$f(c)$ is the cost function of the project. n is the total number of products and c_a and c_b are the cost of the defective product respectively after the project and before the project. $P(fp)$ represents false positives made by models

CRISP - DM



Business information has been given by the company's data science team to group. We defined an expected value framework above with respect to these informations.

First we got information from business department and then we tried to grasp deeply the dataset by using visualization techniques.

Missing value handling, Outlier handling. Converting data types and creating new columns for quality, shifts and datetimes.

We used statistical tests and different models to optimize the problem.

Statistical test will be examined by its own methods. Since we have a classification problem we will use models which is convenient to this problem.

Since we have no intention to use this analysis in real world application we don't have to deploy our models and analysis.

DATA UNDERSTANDING

- ★ Qualitics dataset has *59 columns* and *8491 rows* obtained from sensor values such as Surec1_Onay_Tarihi, Surec2_Tarihi, Surec1_Baslama_tarihi, FazK_dk, FazS_Basinci_Mean, Kalite, MAKINE, K16...46.

- ★ There are 3 columns which all rows are null values: K2, K2_Tarih, K4

- ★ Some columns also have null values which will be handling in data preparation.

```
RangeIndex: 8491 entries, 0 to 8490
Data columns (total 70 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Surec1_Onay_Tarihi     8491 non-null   datetime64
 1   K2                     1 non-null      float64
 2   K2_TARIH               1 non-null      object
 3   K4                     1 non-null      float64
 4   Surec2_Tarihi          7573 non-null   datetime64
 5   Kalite_Kontrol_Tarihi  8491 non-null   datetime64
 6   MAKINE                 8491 non-null   object
 7   Kalite                 8491 non-null   int64
 8   Surec1_Bitis_Tarihi    8491 non-null   datetime64
 9   Surec1_Baslama_Tarihi  8491 non-null   datetime64
 10  PART_NO                8491 non-null   int64
 11  fazK_dk                8482 non-null   float64
 12  FazS_dk                8480 non-null   float64
 13  FazD_dk                8484 non-null   float64
 14  FazB_dk                8484 non-null   float64
 15  FazS_Basinci_Mean      8471 non-null   float64
 16  FazS_Basinci_Stdev     8471 non-null   float64
 17  FazB_Basinci_Max       8484 non-null   float64
 18  FazK_Basinci_Last      8472 non-null   float64
 19  FazD_Basinci_Last      8484 non-null   float64
 20  K17_K16_Mesafe         6984 non-null   float64
 21  K16                    6984 non-null   float64
 22  K18                    6984 non-null   float64
```

DATA PREPARATION

Target Value

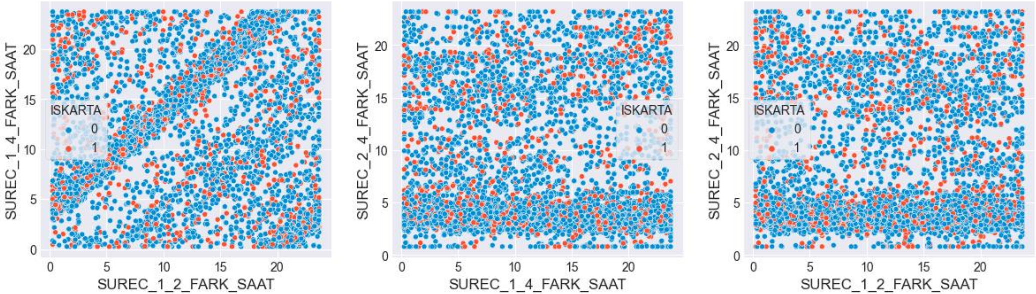
- ★ In Kalite column: 10, 11 shows disqualified items, and 2,1 show qualified ones.
- ★ We create ISKARTA column as 1 is the defective and 0 is the quality product.

Datetime Values

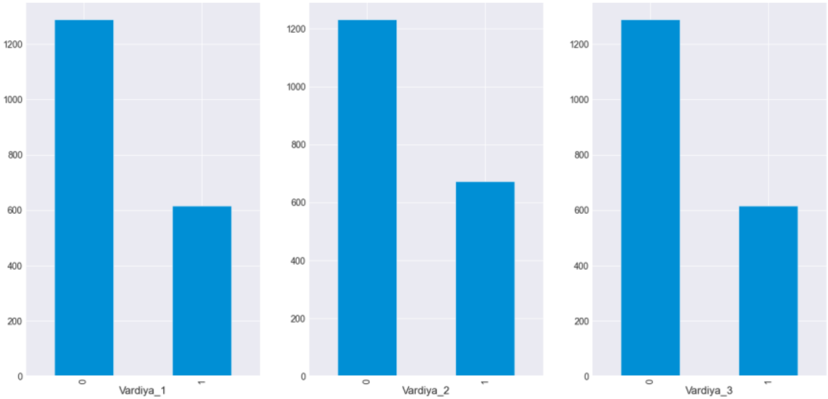
- ★ We convert data types of values ("Surec1_Onay_Tarihi", "Surec2_Tarihi", "Kalite_Kontrol_Tarihi", "Surec1_Bitis_Tarihi", "Surec1_Baslama_Tarihi", "Surec4_Baslangic", "Surec4_Bitis") to datetime types.

DATA PREPARATION Feature Extraction

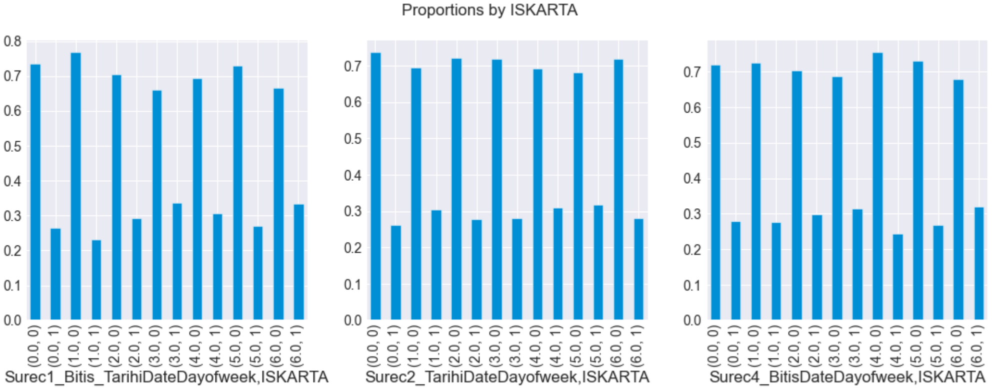
We determine the differences as hours between phases.



Number of products vs. Shifts



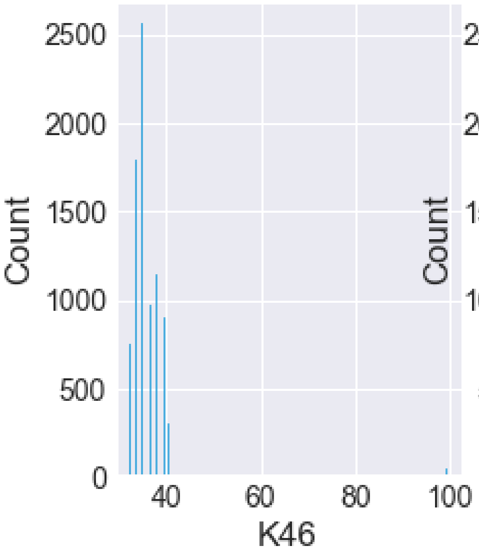
We create features as 'month', 'day', 'dayofweek', 'dayofyear' for each phase

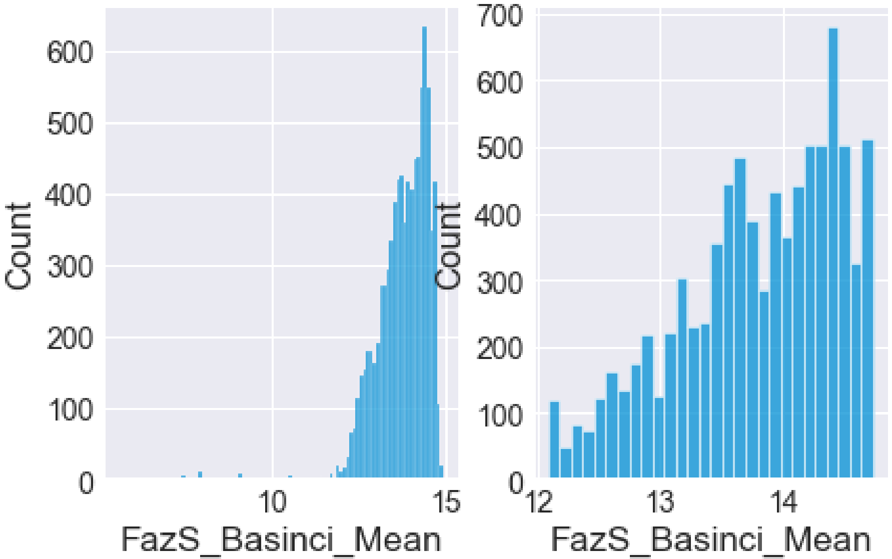


DATA PREPARATION

Outlier Handling

Since there was very few data to begin with, it didn’t seem logical to drop the outliers instead preserving the distribution outlier values are modified to be placed at the 1-99% boundaries.





DATA PREPARATION

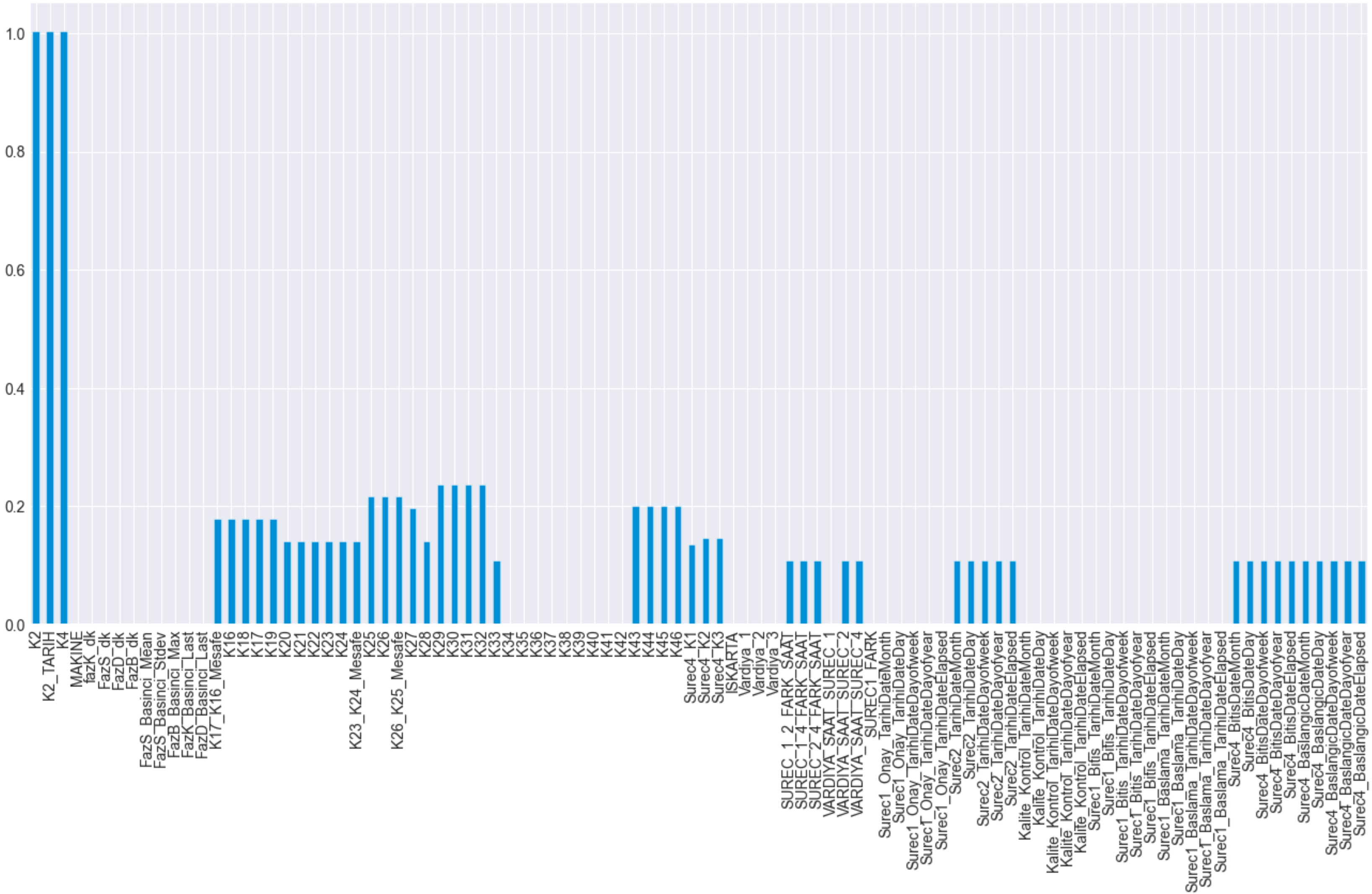
Drop Unused Columns

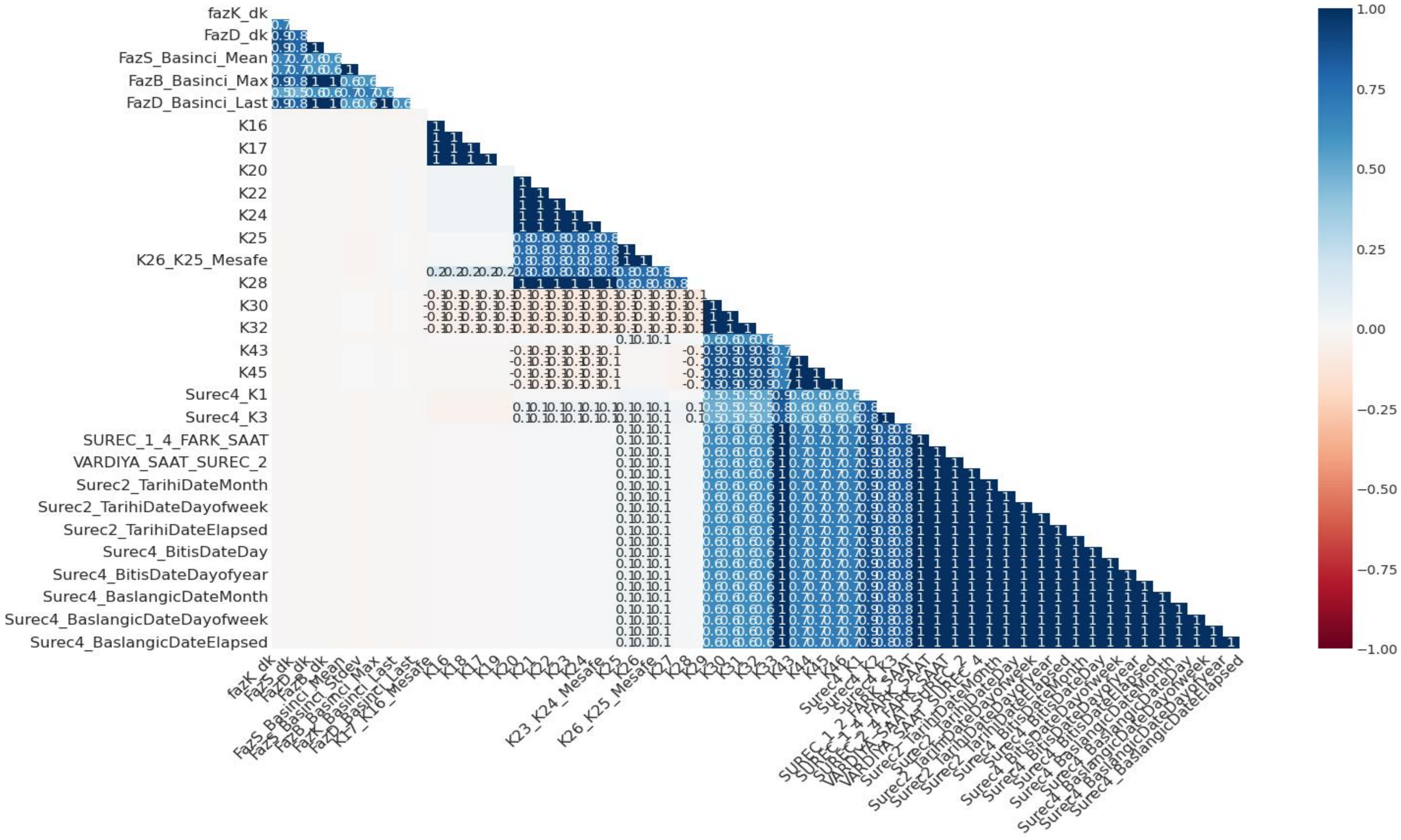
★ Kalite and PART_No columns are dropped.

Missing Value Handling

★ Some of the missing values are seen together. This relationship will be tagged a value between 0 and 1 on the graphically.

★ Rates of missing values are examined for each feature. Columns having higher rate then 95% are dropped.


1.0
0.8
0.6
0.4
0.2
0.0
K2
K2_TARIH
K4
MAKINE
fazK_dk
FazS_dk
FazD_dk
FazB_dk
FazS_Basinci_Mean
FazS_Basinci_Stdev
FazB_Basinci_Max
FazK_Basinci_Last
FazD_Basinci_Last
K17_K16_Mesafe
K16
K18
K17
K19
K20
K21
K22
K23
K24
K23_K24_Mesafe
K25
K26
K26_K25_Mesafe
K27
K28
K29
K30
K31
K32
K33
K34
K35
K36
K37
K38
K39
K40
K41
K42
K43
K44
K45
K46
Surec4_K1
Surec4_K2
Surec4_K3
ISKARTA
Vardiya_1
Vardiya_2
Vardiya_3
SUREC_1_2_FARK_SAAT
SUREC_1_4_FARK_SAAT
SUREC_2_4_FARK_SAAT
VARDIYA_SAAT_SUREC_1
VARDIYA_SAAT_SUREC_2
VARDIYA_SAAT_SUREC_4
SUREC1_FARK
Surec1_Onay_TarihiDateMonth
Surec1_Onay_TarihiDateDay
Surec1_Onay_TarihiDateDayofweek
Surec1_Onay_TarihiDateDayofyear
Surec1_Onay_TarihiDateElapsed
Surec2_TarihiDateMonth
Surec2_TarihiDateDay
Surec2_TarihiDateDayofweek
Surec2_TarihiDateDayofyear
Surec2_TarihiDateElapsed
Kalite_Kontrol_TarihiDateMonth
Kalite_Kontrol_TarihiDateDay
Kalite_Kontrol_TarihiDateDayofweek
Kalite_Kontrol_TarihiDateDayofyear
Kalite_Kontrol_TarihiDateElapsed
Surec1_Bitis_TarihiDateMonth
Surec1_Bitis_TarihiDateDay
Surec1_Bitis_TarihiDateDayofweek
Surec1_Bitis_TarihiDateDayofyear
Surec1_Bitis_TarihiDateElapsed
Surec1_Baslama_TarihiDateMonth
Surec1_Baslama_TarihiDateDay
Surec1_Baslama_TarihiDateDayofweek
Surec1_Baslama_TarihiDateDayofyear
Surec1_Baslama_TarihiDateElapsed
Surec4_BitisDateMonth
Surec4_BitisDateDay
Surec4_BitisDateDayofweek
Surec4_BitisDateDayofyear
Surec4_BitisDateElapsed
Surec4_BaslangicDateMonth
Surec4_BaslangicDateDay
Surec4_BaslangicDateDayofweek
Surec4_BaslangicDateDayofyear
Surec4_BaslangicDateElapsed

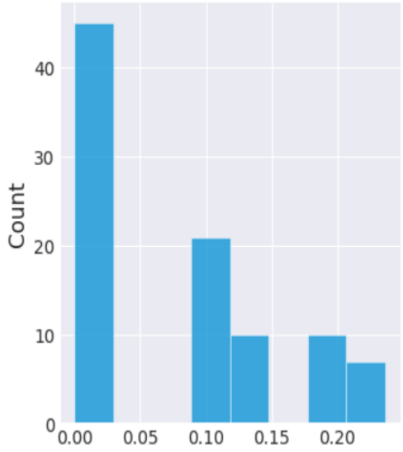

fazK_dk
FazD_dk
FazS_Basinci_Mean
FazB_Basinci_Max
FazD_Basinci_Last
K16
K17
K20
K22
K24
K25
K26_K25_Mesafe
K28
K30
K32
K43
K45
Surec4_K1
Surec4_K3
SUREC_1_4_FARK_SAAT
VARDIYA_SAAT_SUREC_2
Surec2_TarihiDateMonth
Surec2_TarihiDateDayofweek
Surec2_TarihiDateElapsed
Surec4_BitisDateDay
Surec4_BitisDateDayofyear
Surec4_BaslangicDateMonth
Surec4_BaslangicDateDayofweek
Surec4_BaslangicDateElapsed
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

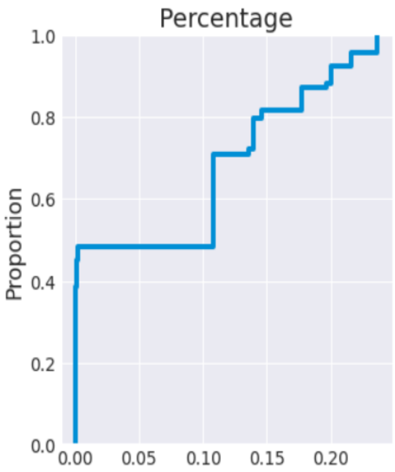
DATA PREPARATION

Missing Value Handling

SimpleImputer() is used to fill the missing values with “median” values. Other functions such as “mean” and “most frequent” are also tested yielding similar results.

Rates of Missing Values





DATA PREPARATION

Encoding Categorical Variables

Original dataset had only 2 categorical variable. After feature extraction, some other categorical features are derived like day of week, hour and month.

For columns which consists from 2 values we used Label Encoder. Otherwise, One Hot Encoding was used to convert data.

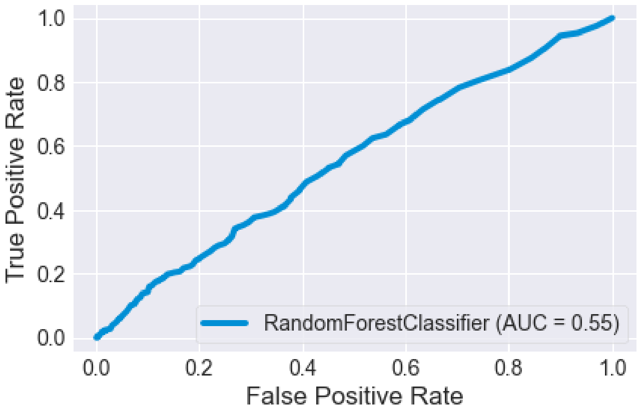
Dummy Variable Trap are taken into account.

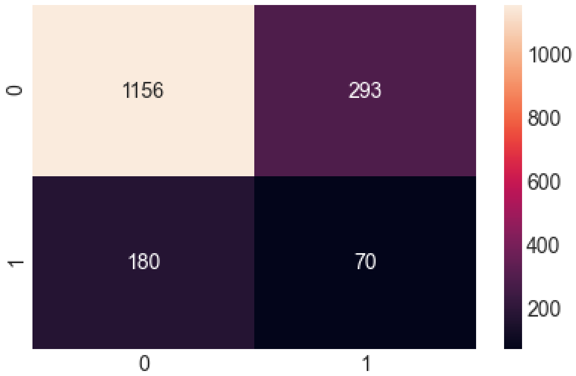
Train-Test Splits

Different train-test splitting techniques used on modelling phase. First, all columns included on splits, then only significant columns included for retrained models. For autoencoder model a special way preferred.

MODELLING

Random Forest Classifier

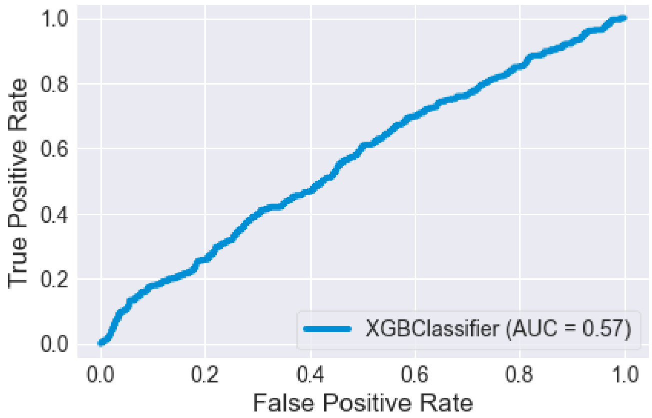


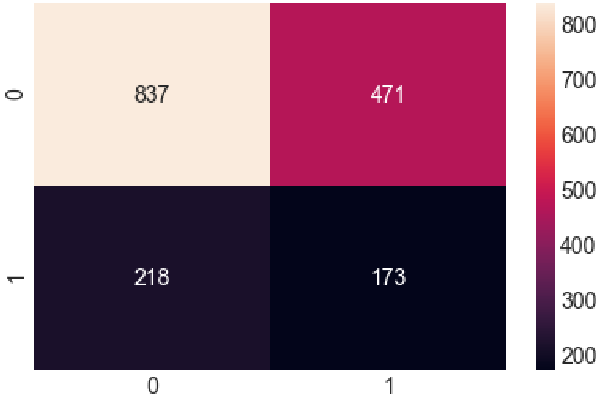


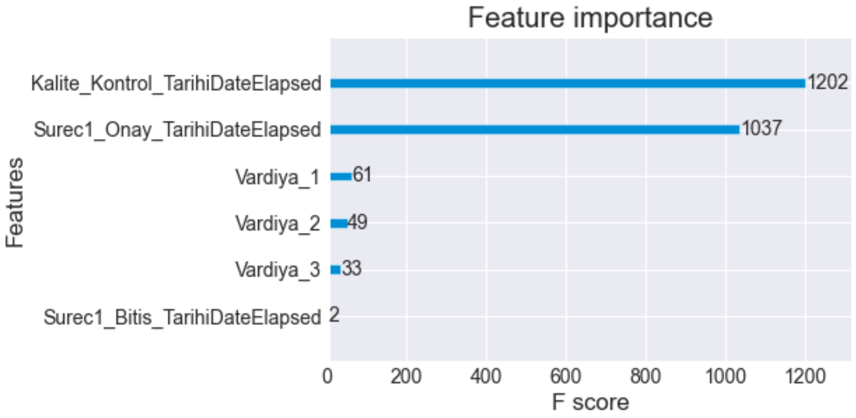
	precision	recall	f1-score	support
0	0.79	0.89	0.83	1327
1	0.27	0.15	0.19	372
accuracy			0.72	1699
macro avg	0.53	0.52	0.51	1699
weighted avg	0.67	0.72	0.69	1699

MODELLING

XGBoost Classifier



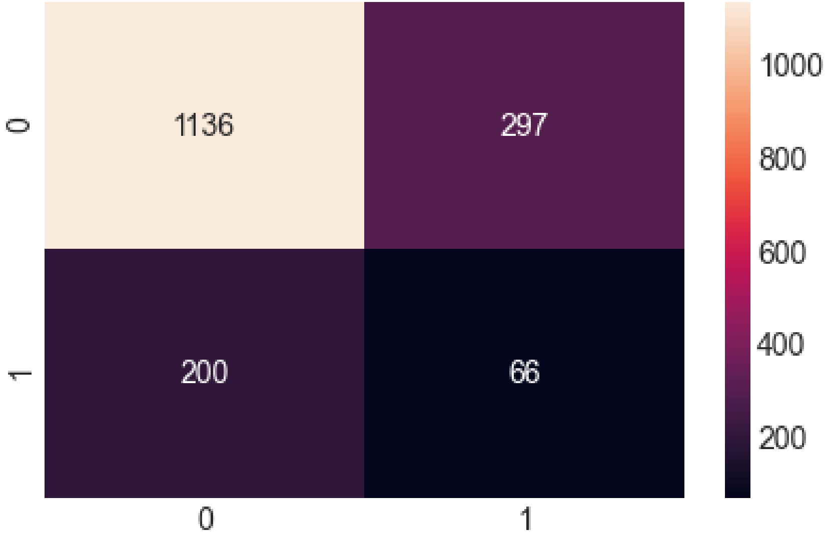




	precision	recall	f1-score	support
0	0.79	0.64	0.71	1308
1	0.27	0.44	0.33	391
accuracy			0.59	1699
macro avg	0.53	0.54	0.52	1699
weighted avg	0.67	0.59	0.62	1699

MODELLING

Neural Networks

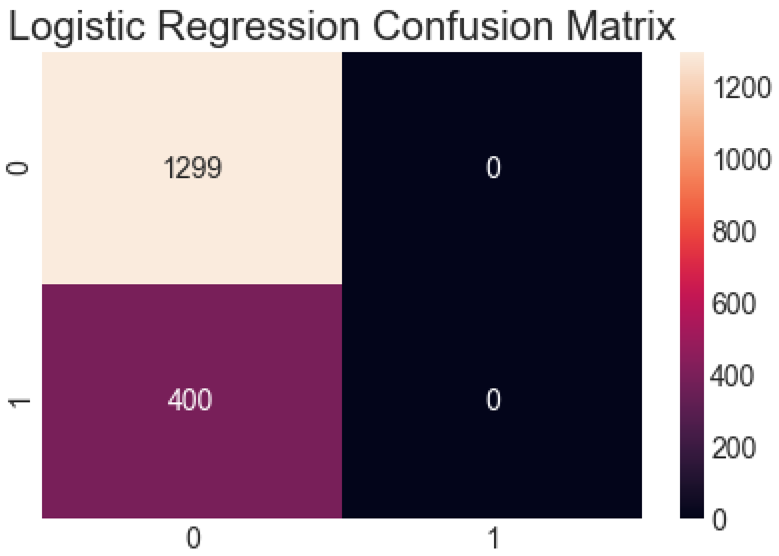


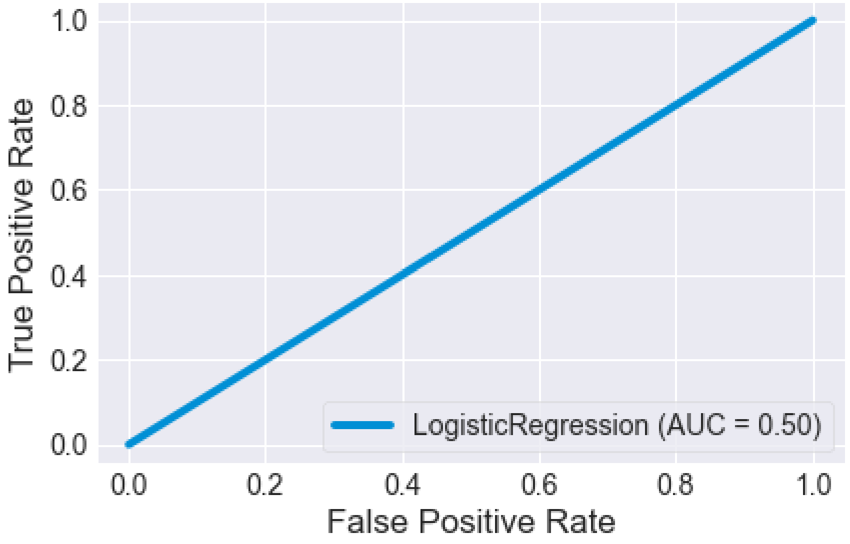
	precision	recall	f1-score	support
0	0.79	0.85	0.82	1336
1	0.25	0.18	0.21	363
accuracy			0.71	1699
macro avg	0.52	0.52	0.52	1699
weighted avg	0.68	0.71	0.69	1699

MODELLING

Logistic Regression

	precision	recall	f1-score	support
0	0.76	1.00	0.87	1299
1	0.00	0.00	0.00	400
accuracy			0.76	1699
macro avg	0.38	0.50	0.43	1699
weighted avg	0.58	0.76	0.66	1699

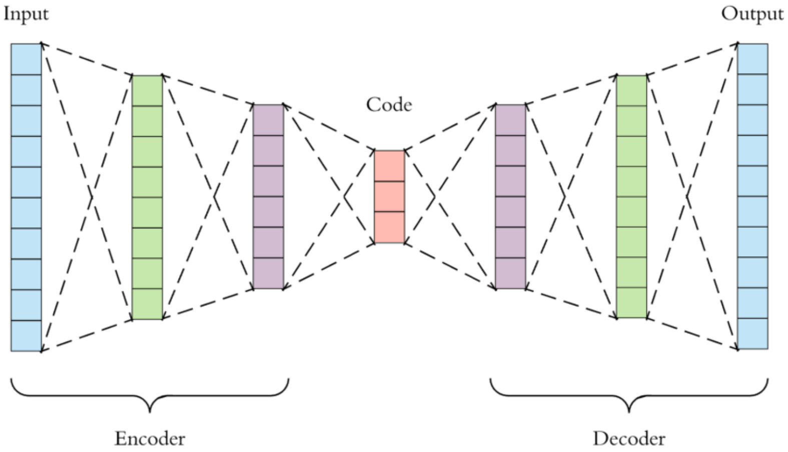




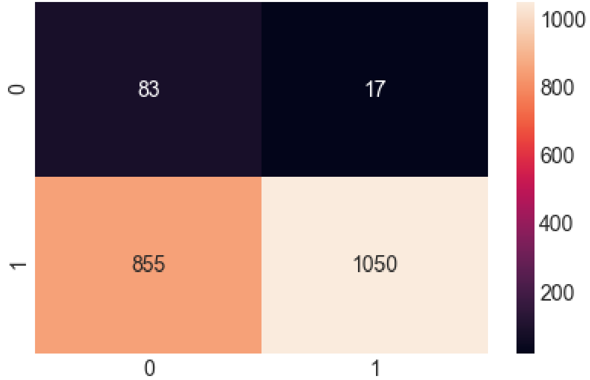
Modelling

One Class Learning with AutoEncoder

Since there are too few data to train an autoencoder, only 100 negative samples were taken for test and used the rest of data to train. But also getting 1000 test samples did not change the results much.



	precision	recall	f1-score	support
0	0.09	0.83	0.16	100
1	0.98	0.55	0.71	1905
accuracy			0.57	2005
macro avg	0.54	0.69	0.43	2005
weighted	0.94	0.57	0.68	2005



AutoEncoder Image Source : https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798

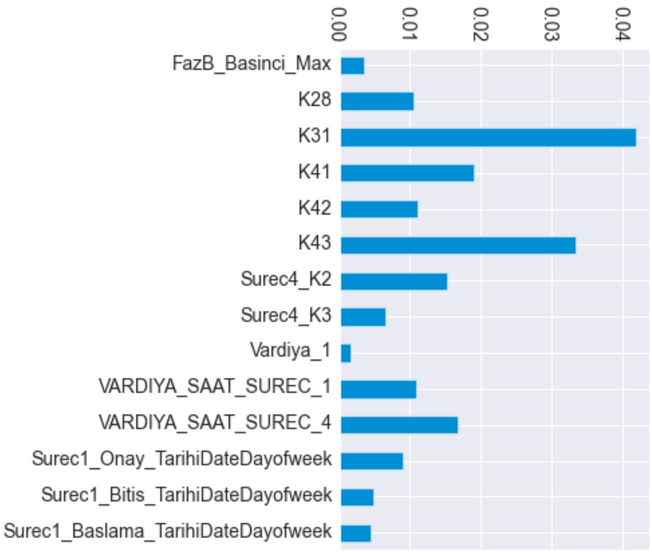
Modelling

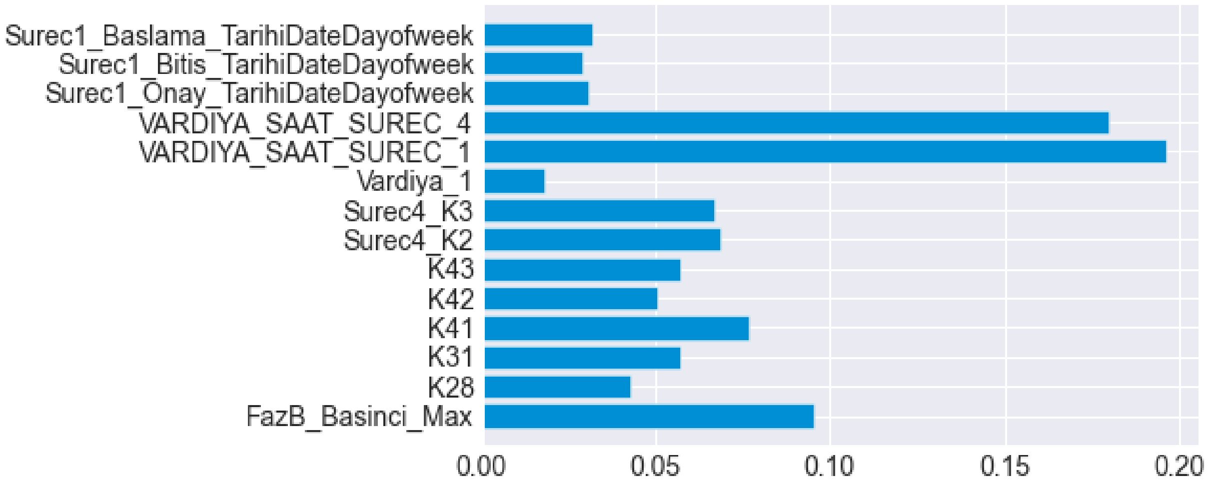
MODEL COMPARISON

Models	Classes	AUC	ACCURACY	PRECISION	RECALL	F1-SCORE
Random Forest	0	0.55	0.72	0.79	0.89	0.83
	1			0.27	0.15	0.19
XGB Classifier	0	0.57	0.59	0.79	0.64	0.71
	1			0.27	0.44	0.33
Neural Network	0	-	0.71	0.79	0.85	0.82
	1			0.25	0.18	0.21
Logistic Regression	0	0.50	0.76	0.76	1.00	0.87
	1			0.00	0.00	0.00
AutoEncoder	0	-	0.57	0.09	0.83	0.16
	1			0.98	0.55	0.71

ANALYSIS

P values of each column are investigated in data according to target variable. Some of the variables which are created on feature engineering section are considered meaningful wrt their p values. Also, testing is seen to be in compliance with the featured importances extracted from modelling.





ANALYSIS

Review of Hypothesis Testing

Two-Sample T-Test and CI: FazB_0, FazB_1

Method

μ_1: mean of FazB_0
μ_2: mean of FazB_1
Difference: $\mu_1 - \mu_2$

Equal variances are not assumed for this analysis.

Descriptive Statistics

Sample	N	Mean	StDev	SE Mean
FazB_0	5897	7.178	0.268	0.0035
FazB_1	1676	7.199	0.254	0.0062

Estimation for Difference

Difference	95% CI for Difference
-0.02158	(-0.03553, -0.00764)

Test

Null hypothesis $H_0: \mu_1 - \mu_2 = 0$
Alternative hypothesis $H_1: \mu_1 - \mu_2 \neq 0$

T-Value	DF	P-Value
-3.03	2826	0.002

Two-Sample T-Test and CI: Var_Sa_S1_0, Var_Sa_S1_1

Method

μ_1: mean of Var_Sa_S1_0
μ_2: mean of Var_Sa_S1_1
Difference: $\mu_1 - \mu_2$

Equal variances are not assumed for this analysis.

Descriptive Statistics

Sample	N	Mean	StDev	SE Mean
Var_Sa_S1_0	5897	11.00	6.77	0.088
Var_Sa_S1_1	1905	11.46	6.70	0.15

Estimation for Difference

Difference	95% CI for Difference
-0.461	(-0.808, -0.114)

Test

Null hypothesis $H_0: \mu_1 - \mu_2 = 0$
Alternative hypothesis $H_1: \mu_1 - \mu_2 \neq 0$

T-Value	DF	P-Value
-2.60	3253	0.009

Conclusion

In this project, we examined whether it is possible to model and predict defective products before the production process ends. Also, it is important to find which phases of production causes defective products. Hence, the production line could be redesigned and defective rates could be decreased.

The final conclusion of this analysis is, since there are lots of unmeasured and randomized affects on product phase, it is hard to predict defective products before process ends. But some phases of production has affects on outcome via probility of being defective. These findings have been presented. By redesigning production phase with respect to findings, could decrease defective product rate.

Also there is some unofficial and non-significant results including affects like working at Sunday and Faz_D_Basinci_Last.

This project has higher cost than acceptable, with respect to our Expected Value formula. Because all models comes with high rate of false positives.

Interpretation of the analysis is profoundly bond to the knowledge about production cycle.

THANKS FOR LISTENING

Project and Dataset:
https://drive.google.com/drive/folders/1C0Ez6EguomlciwFHzrb6K_hesaDxh0xH?usp=sharing