# Predicting Sepsis on the PredictEM Masked Blutkulturen Dataset - May

Bachelor Thesis by Yannick Müller

# What I did

1. K-Fold Validation: improvement AUC_ROC from 0.64 to 0.67 as whole dataset was trained
2. Boolean Variables: minor improvement
3. Feature Selection: minor improvement
4. ROC_AUC: Implemented
5. Ensemble: got worse
6. One-Hot-Encoding
7. Hyperparameter tuning: minor improvement, still working on it
8. Created Minimal Example

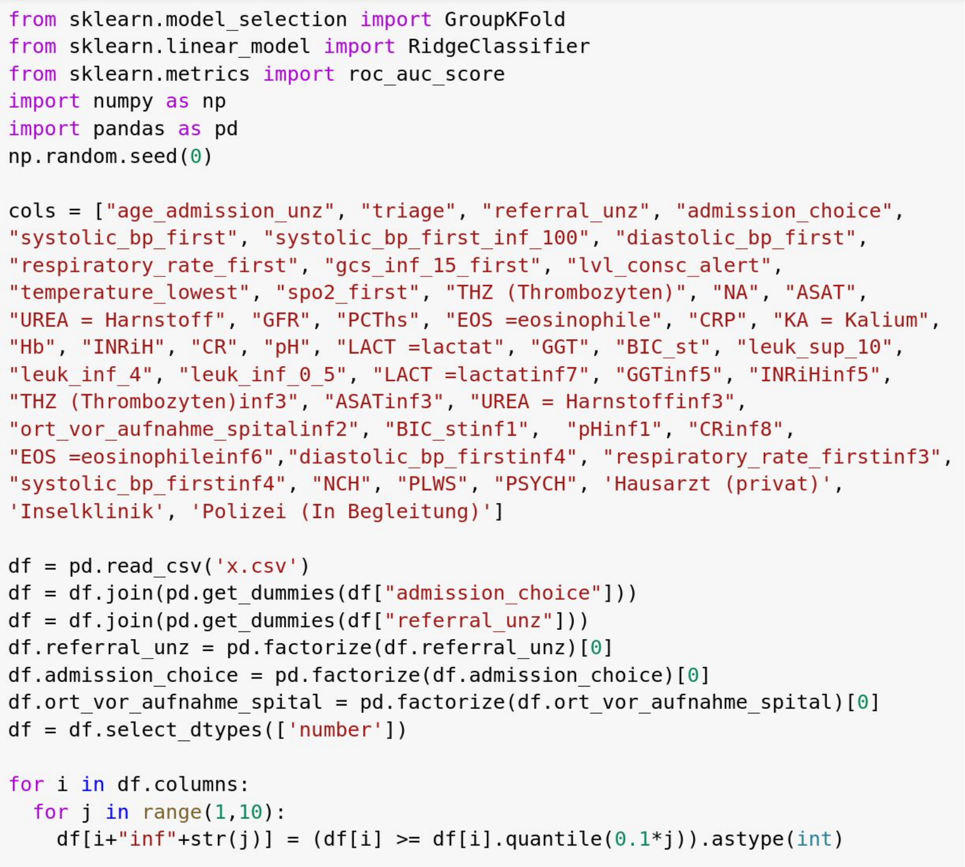
# Improved 26 Features

LACT =lactat, LACT =lactat inf quant 0.7, CRP, GGT, GGT inf median, INRiH inf median, INRiH, THZ (Thrombozyten), THZ (Thrombozyten) inf quant 0.3, ASAT inf quant 0.3, UREA = Harnstoff, UREA = Harnstoff inf quant 0.3, ort_vor_aufnahme_spital: anderes Spital, BIC_VB, BIC_st, BIC_st inf quant 0.1, lvl_consc_alert, pH inf quant 0.1, CR inf quant 0.8, EOS =eosinophile inf quant 0.6, age_admission_unz, diastolic_bp_first inf quant 0.4, respiratory_rate_first inf quant 0.3, systolic_bp_first inf quant 0.4, Hb, PCThs

# Predicting blood culture positive

| Average Score 8-Fold Validation Set | F1 Score | AUC_ROC |
|---|---|---|
| Linear Regression | 0.45 | 0.67 |
| Ridge Classifier | 0.45 | 0.67 |
| Ridge Regression | 0.45 | 0.67 |
| Bayesian Ridge | 0.45 | 0.67 |
| XGB Classifier | 0.43 | 0.66 |
| Gaussian Naive Bayes | 0.43 | 0.65 |

# Preprocessing Example



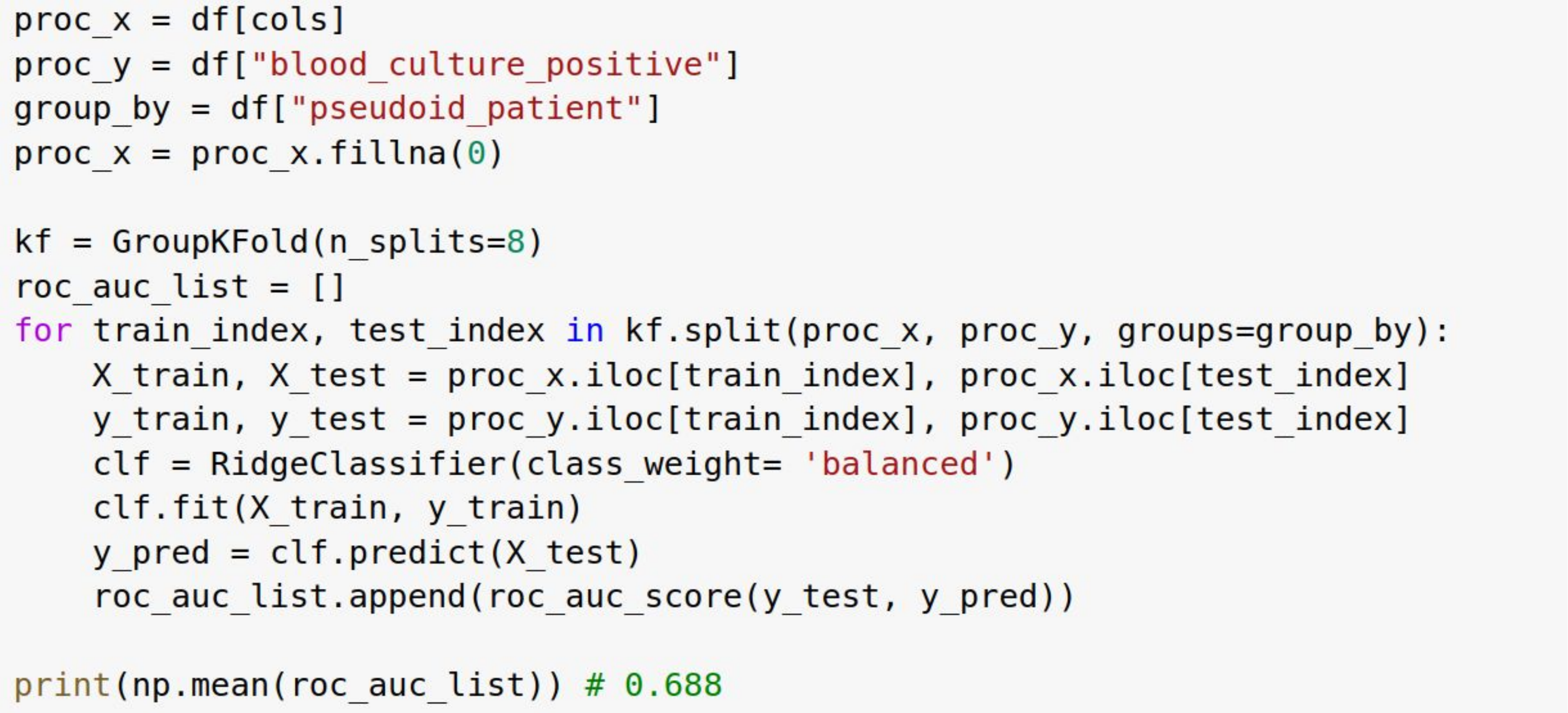
```python
from sklearn.model_selection import GroupKFold
from sklearn.linear_model import RidgeClassifier
from sklearn.metrics import roc_auc_score
import numpy as np
import pandas as pd
np.random.seed(0)

cols = ["age_admission_unz", "triage", "referral_unz", "admission_choice",
"systolic_bp_first", "systolic_bp_first_inf_100", "diastolic_bp_first",
"respiratory_rate_first", "gcs_inf_15_first", "lvl_consc_alert",
"temperature_lowest", "spo2_first", "THZ (Thrombozyten)", "NA", "ASAT",
"UREA = Harnstoff", "GFR", "PCThs", "EOS =eosinophile", "CRP", "KA = Kalium",
"Hb", "INRiH", "CR", "pH", "LACT =lactat", "GGT", "BIC_st", "leuk_sup_10",
"leuk_inf_4", "leuk_inf_0_5", "LACT =lactatinf7", "GGTinf5", "INRiHinf5",
"THZ (Thrombozyten)inf3", "ASATinf3", "UREA = Harnstoffinf3",
"ort_vor_aufnahme_spitalinf2", "BIC_stinf1",  "pHinf1", "CRinf8",
"EOS =eosinophileinf6","diastolic_bp_firstinf4", "respiratory_rate_firstinf3",
"systolic_bp_firstinf4", "NCH", "PLWS", "PSYCH", 'Hausarzt (privat)',
'Inselklinik', 'Polizei (In Begleitung)']

df = pd.read_csv('x.csv')
df = df.join(pd.get_dummies(df["admission_choice"]))
df = df.join(pd.get_dummies(df["referral_unz"]))
df.referral_unz = pd.factorize(df.referral_unz)[0]
df.admission_choice = pd.factorize(df.admission_choice)[0]
df.ort_vor_aufnahme_spital = pd.factorize(df.ort_vor_aufnahme_spital)[0]
df = df.select_dtypes(['number'])

for i in df.columns:
  for j in range(1,10):
    df[i+"inf"+str(j)] = (df[i] >= df[i].quantile(0.1*j)).astype(int)
```

# Example Ridge Classifier



```
proc_x = df[cols]
proc_y = df["blood_culture_positive"]
group_by = df["pseudoid_patient"]
proc_x = proc_x.fillna(0)

kf = GroupKFold(n_splits=8)
roc_auc_list = []
for train_index, test_index in kf.split(proc_x, proc_y, groups=group_by):
    X_train, X_test = proc_x.iloc[train_index], proc_x.iloc[test_index]
    y_train, y_test = proc_y.iloc[train_index], proc_y.iloc[test_index]
    clf = RidgeClassifier(class_weight= 'balanced')
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)
    roc_auc_list.append(roc_auc_score(y_test, y_pred))

print(np.mean(roc_auc_list)) # 0.688
```

# Effectiveness different elements

| Element | Improvement ROC_AUC |
|---|---|
| Feature Selection | 0.014 |
| Binary threshold encoding | 0.023 |
| One-Hot-Encoding | 0.017 |
| 8-fold validation instead of 80/20 Split | 0.006 |
| No scaling instead of Normalization | 0.026 |
| No scaling instead of StandardScaler | 0.001 |
| Ridge Classification instead of Linear Regression | 0.004 |
| Missing values replace with constant instead of mean | 0.042 |
| Class weight balanced instead of weight 1 | 0.157 |

# Disallowed

| Element | Improvement ROC_AUC |
|---|---|
| Regression instead of Classification | 0.052 |
| Shuffle instead of group split | 0.003 |

If it was allowed a AUC_ROC score of 0.727 could be achieved

# Up Next

1. Polynomial functions on features
2. More advanced neural network
3. Fourier Transform
4. Pretrained Net for Medical Data and then fine tune on this dataset
5. Sklearn GridSearchCV
6. Min, Max, Mean, Messungen beim Patienten oder falls NaN immer die alten
7. Bidirectional RNN

# Questions

1. ROC_AUC Score from regression between 0 and 1 or classification between 0 and 1. Regression 0.72, Classification 0.67
2. Can I do K-Fold Validation on the whole dataset or do I still need a 20% Test set?
3. What can I do to further improve the model?
4. Do you have any improvements to the minimal example