Hilton Prediction Challenge

AGENDA

Background

HILTON PREDICTION CHALLENGE

BACKGROUND

HR Analytics

Workforce Analytics

People Analytics

Talent Analytics

Human Capital Management


Richard Branson
@richardbranson
Train people well enough so they can leave, treat them well enough so they don't want to
virgin.com
Richard Branson's blog on business and advocacy | Virgin
Richard Branson's blog - where he shares his thoughts on business, Virgin news, leadership, life, family and the globa...

What Leaders Say

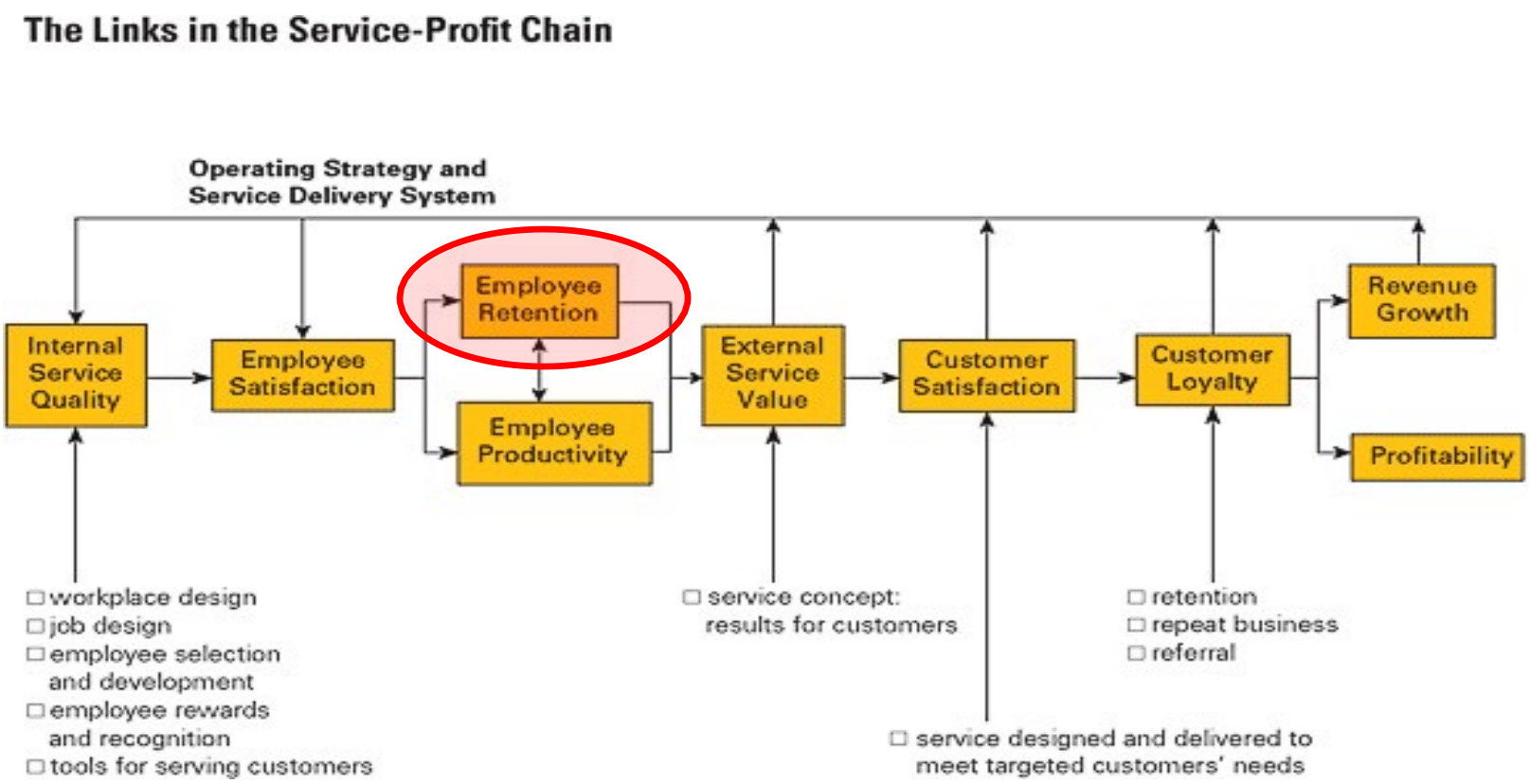
"If you ever get lucky enough to be hiring people, make sure you're hiring people that not only you can teach, but make sure you're hiring people who are also going to teach you things."

- Jeff Bezos

"The competition to hire the best will increase in the years ahead. Companies that give extra flexibility to their employees will have the edge in this area."

- Bill Gates

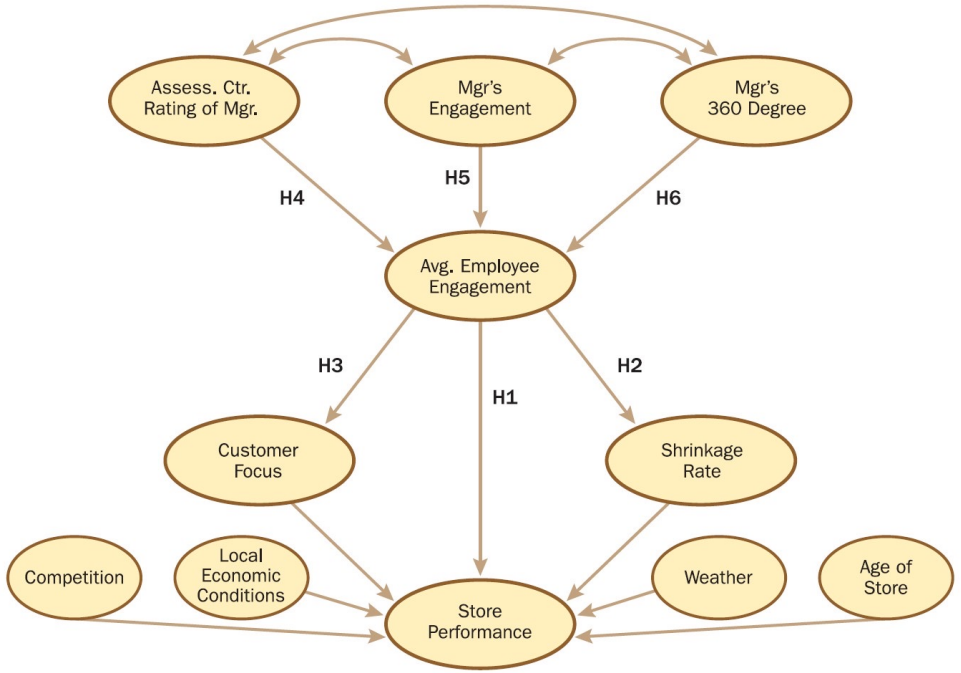
BACKGROUND


The Links in the Service-Profit Chain
Operating Strategy and Service Delivery System
Internal Service Quality
Employee Satisfaction
Employee Retention
Employee Productivity
External Service Value
Customer Satisfaction
Customer Loyalty
Revenue Growth
Profitability
workplace design
job design
employee selection and development
employee rewards and recognition
tools for serving customers
service concept: results for customers
service designed and delivered to meet targeted customers' needs
retention
repeat business
referral

HILTON PREDICTION CHALLENGE

BACKGROUND

EXHIBIT 1: LOWE'S FIRST STORE MODEL BLUEPRINT



The initial Lowe's store model blueprint and roadmap for the final models represented initial hypotheses from the key stakeholders about how the data would interact in the model.

Code and Concepts

Background | **Code and Concepts** | Kaggle

Original Notebook

- Import Packages
- Loading the Hilton Data
- Train a Classifier
- Evaluate the Classifier
  - Confusion Matrix
  - Performance Metrics
  - ROC Curve
- Apply the Model to the Kaggle Data

HILTON PREDICTION CHALLENGE

CODE & CONCEPTS – Loading the Data

2. Load Data

[2]:
```
trainInput = pd.read_csv("Data/hilton_2023_train.csv")
testInput = pd.read_csv("Data/hilton_2023_test.csv")
```

[3]:
```
pd.DataFrame(trainInput.columns).to_csv("cols.csv", index = False)
```

[4]:
```
trainData = trainInput.drop(columns = 'IntentToStayHighLow')
trainLabels = LabelEncoder().fit_transform(trainInput.IntentToStayHighLow)

testData = testInput.drop(columns = 'IntentToStayHighLow')
testLabels = LabelEncoder().fit_transform(testInput.IntentToStayHighLow)
```

CODE & CONCEPTS – Training the Model

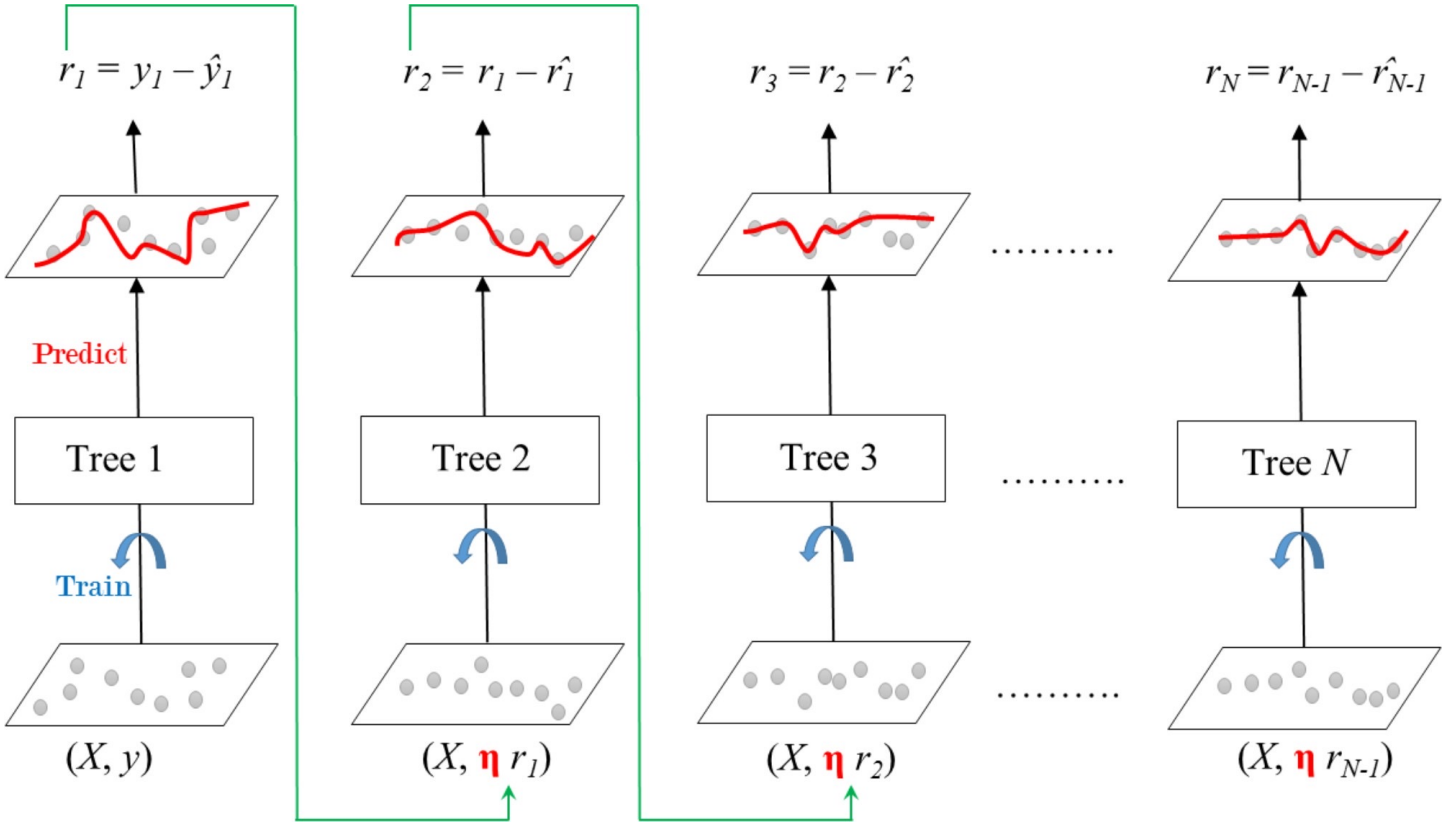
3. Train a XGBoost Classifier

```
# fit the model
clf = XGBClassifier(random_state = 1)
clf.fit(trainData, trainLabels)
```

XGBClassifier

```
XGBClassifier(base_score=0.5, booster='gbtree', callbacks=None,
              colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
              early_stopping_rounds=None, enable_categorical=False,
              eval_metric=None, feature_types=None, gamma=0, gpu_id=-1,
              grow_policy='depthwise', importance_type=None,
              interaction_constraints='', learning_rate=0.300000012,
              max_bin=256, max_cat_threshold=64, max_cat_to_onehot=4,
              max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
              missing=nan, monotone_constraints='()', n_estimators=100,
              n_jobs=0, num_parallel_tree=1, predictor='auto', random_state=1, ..
.)
```

CODE & CONCEPTS – Boosting



CODE & CONCEPTS – Evaluate the Classifier

```
from custom_functions import plot_conf_mat, plot_roc_curve, plot_feature_importance, calculateMetric
```

4.1. Confusion Matrix: ¶

```
plot_conf_mat(clf, # The classifier object
              testData, # The test data set aside for evaluation in train_test_split
              testLabels # Actual labels
             )
```

4.2. Accuracy, Precision, Recall, AUC, and F1:

```
predictedProbabilities = clf.predict_proba(testData)
predictedLabels = clf.predict(testData)
calculateMetricsAndPrint(predictedLabels, predictedProbabilities, testLabels)
```

4.3. ROC Curve:

```
positiveProbabilities = predictedProbabilities[:,1]
```

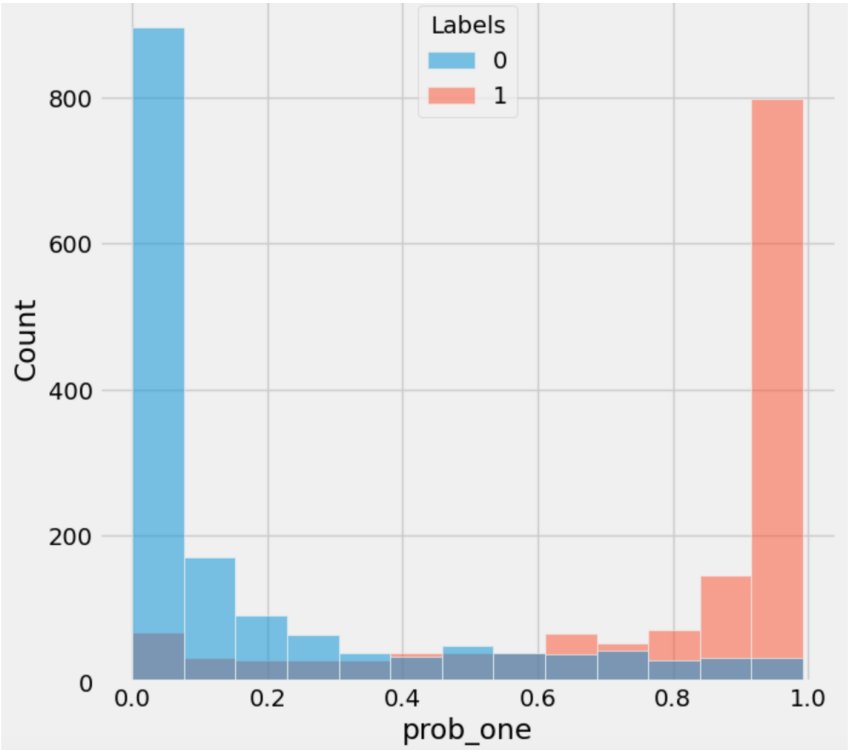
```
plot_roc_curve(testLabels, # Actual labels
               positiveProbabilities, # Prediction scores for the positive class
               pos_label = 1 # Indicate the label that corresponds to the positive class
              )
```

CODE & CONCEPTS – Evaluate the Classifier

4.4. Log-Loss

```
temp = pd.DataFrame(positiveProbabilities)
temp.columns = ["prob_one"]
temp["Labels"] = testLabels

sns.histplot(data=temp, x="prob_one", hue="Labels")
plt.show()
```




```
log_loss(testLabels,positiveProbabilities)
```

```
0.3747093055941757
```

CODE & CONCEPTS – Apply to the Kaggle Data

5. Apply the Model to Kaggle Data:

```
kaggleTest = pd.read_csv("Data/hilton_2024_kaggle.csv")
```

```
kaggleTest['score'] = clf.predict_proba(kaggleTest.drop(columns = 'unique_id'))[:,1]
kaggleTest[['unique_id','score']].to_csv("Data/Kaggle_Submission.csv", index = False)
```

Please submit to https://www.kaggle.com/t/72810a50d9ea40a287e0fdb347a07db9

Each team should make at least three submissions per week from the launch until we close the competition.

How to Improve My Model?

1- Collect more/ better data 

2- Pre-process the data

3- Use a better algorithm (or a set of algorithms)

4- Find better values for the hyperparameters

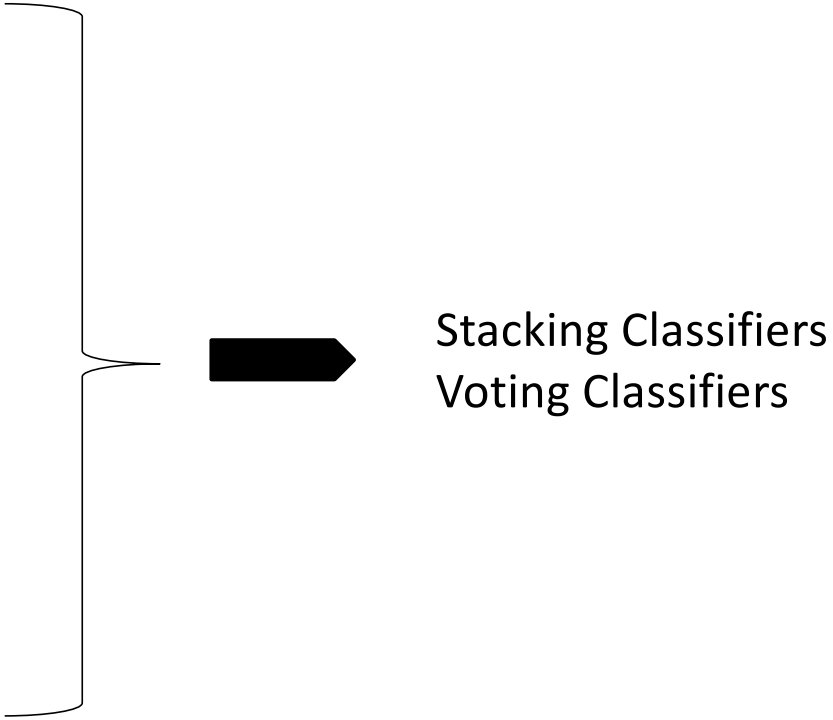
Pre-process the Data

Process	Method	Code
Feat. Select	Boruta	BorutaPy(...)
	sklearn.feature_selection	RFE(...)
Imputation	sklearn.impute	IterativeImputer(...)
Cat. Vars./ Scale Vars.	category_encoders	BaseNEncoder(...)
	sklearn.preprocessing	OneHotEncoder(...) StandardScaler(...)
All Feat. Eng.	feature-engine	MeanMedianImputer(...) OutlierTrimmer(...) MathematicalCombination(...) DropCorrelatedFeatures(...) DecisionTreeDiscretiser(...)

Use a Better Algorithm

Process	Method	Code
Select Algo.	mljar-supervised	AutoML(…)
	pycaret	compare_models(…)

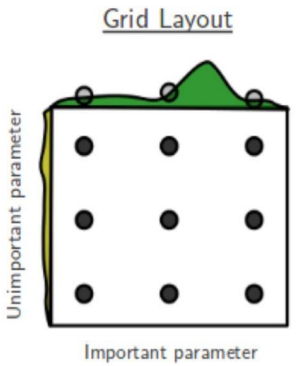
Process	Method	Code
Final Tuned Algo.	xgboost	XGBClassifier(…)
	catboost	CatBoostClassifier(…)
	lightgbm	LGBMClassifier(…)

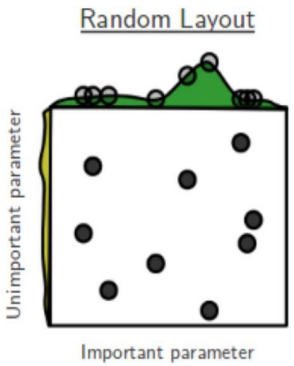


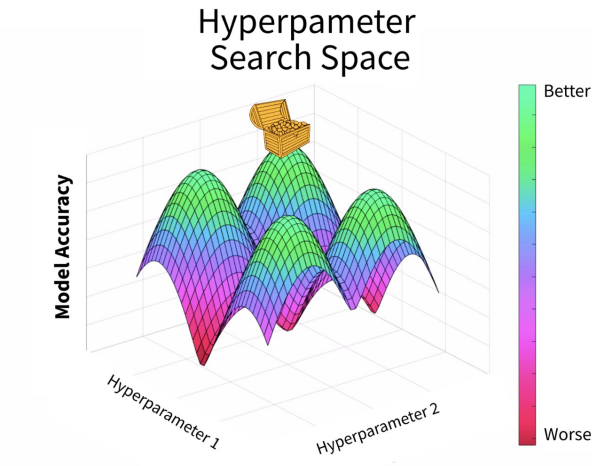
Stacking Classifiers
Voting Classifiers

Find Better Values for the Hyperparameters

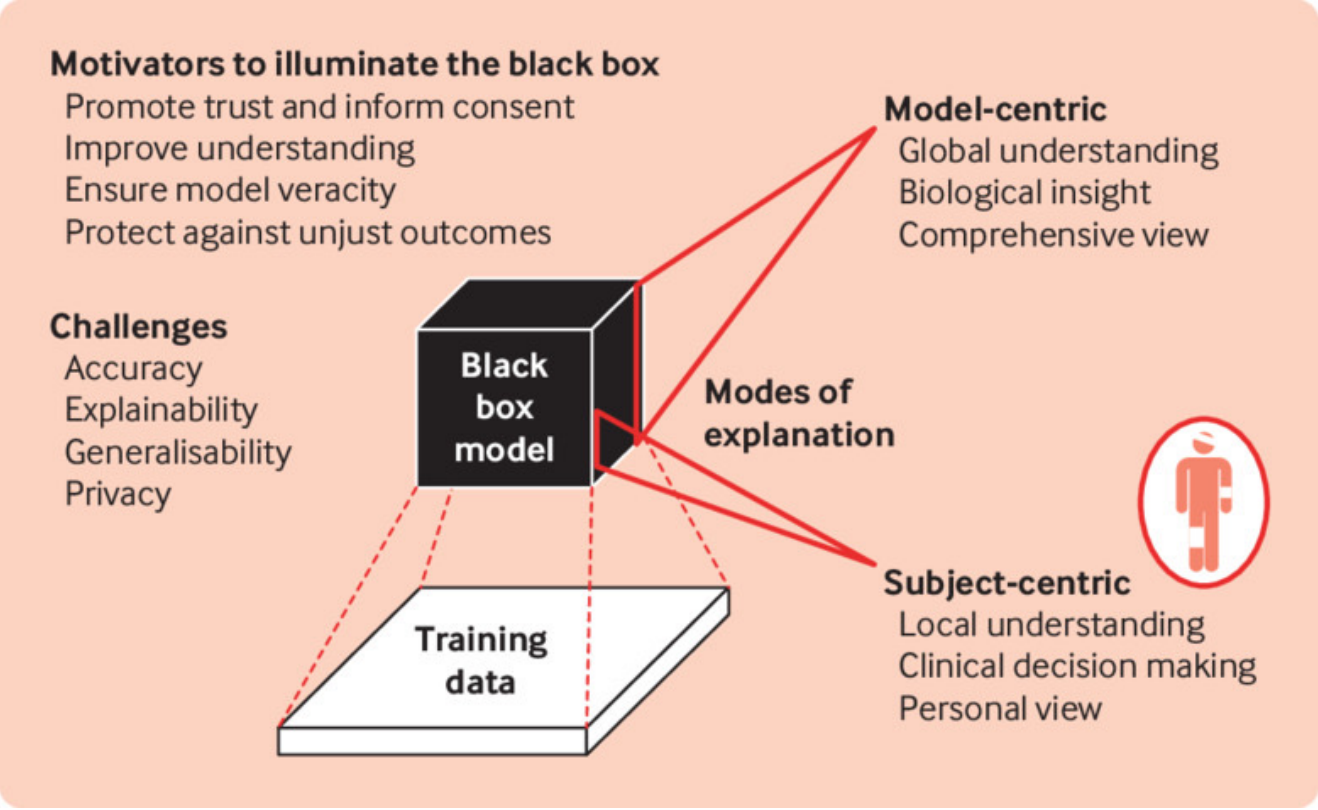
Process	Method	Code
Hyper Param. Tuning	hyperopt	fmin(…)
	sklearn.model_selection	GridSearchCV(…)
		RandomizedSearchCV(…)







CODE & CONCEPTS – How To Enrich My Story?



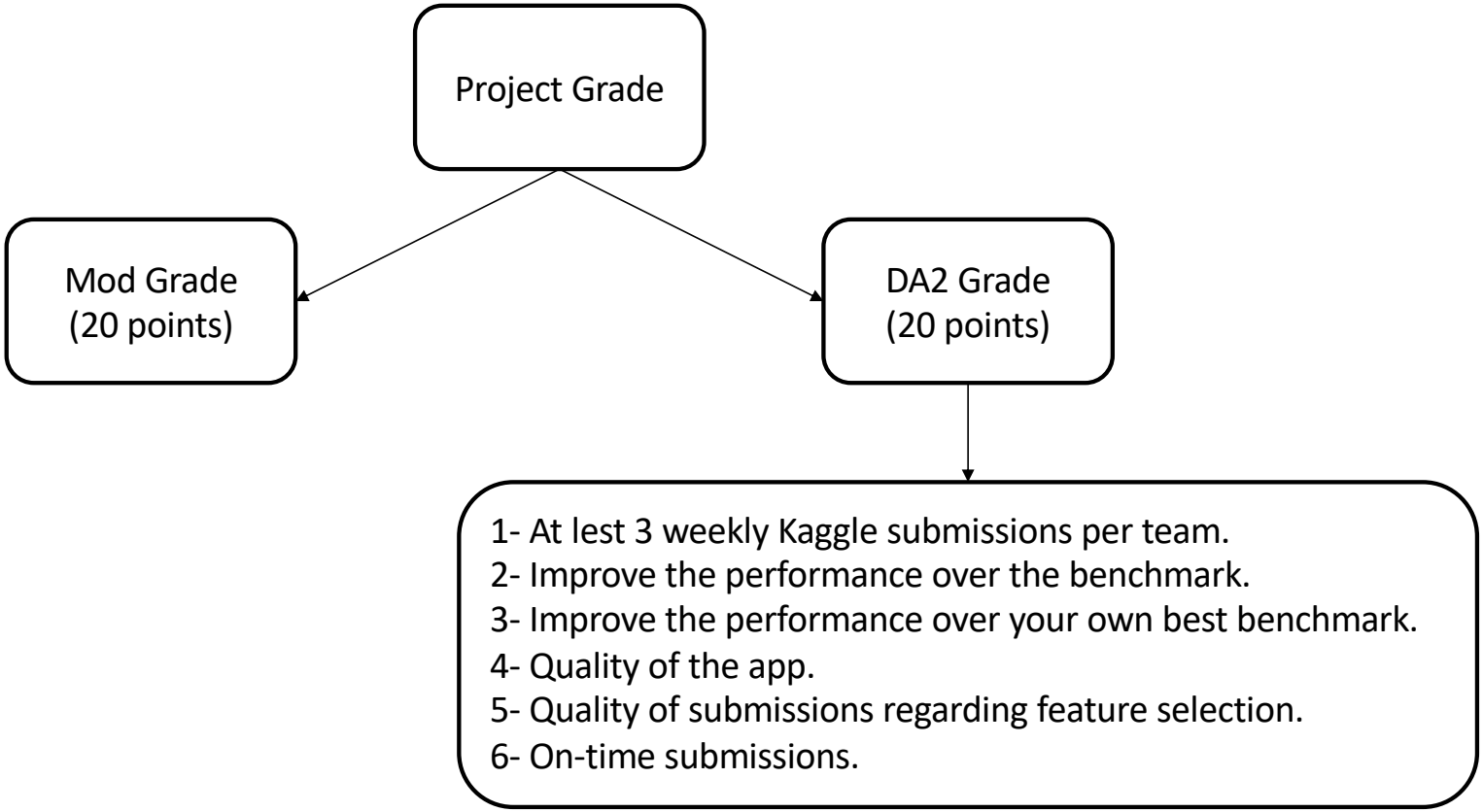
CODE & CONCEPTS – How To Enrich My Story?

Approach Name	Model Centric	Subject Centric
Feature Ranking	plot_importance(clf, …)	
Recursive Feature Engineering (RFE)	RFE(clf, …)	
Boruta	BorutaPy(estimator=clf, …)	
Logistic Regression	sm.Logit(y, X).fit()	
Shap	shap.summary_plot(shapValues, …) shap.plots.scatter(shapValuess[:,"col1"], …)	shap.force_plot(…) shap.plots.waterfall(…)
Lime		explainer.explain_instance(…)
iModels	HSTreeClassifierCV(max_leaf_nodes=6)	
Explainer Dashboard	ClassifierExplainer(clf, testData, testLabels)	ClassifierExplainer(clf, testData, testLabels)

CODE & CONCEPTS – How To Enrich My Story?

- Compare different groups of employees with others
  - Group by FullTimePartTime
  - Group by ManagementLevel
  - Group by …
- Find the most effective strategies for each group of employees.
- Identify potential interactions among predictors. (Hypothetical) examples:
  - WorkLifeBalance is important only if people are happy with management.
  - Engagement is more important for full time employees only.

Project Grade



Due Dates and Deliverables

Competition Kick-off: 01/23

Checkpoint	Date (due 11:59 PM ET)	Deliverables
1	01/30	At least 3 submissions to Kaggle
2	02/06	At least 3 submissions to Kaggle, Beat Kaggle benchmark
3	02/13	At least 3 submissions to Kaggle, Identify the top 30 features
4	02/20	At least 3 submissions to Kaggle, Identify the top 20 features
5	02/27	At least 3 submissions to Kaggle, Beat your own top submission, Select features for the app
6	03/05	At least 3 submissions to Kaggle, App prototype (your MVP- Takes several features as input and displays the predicted score as output.)
7	03/08	Final version of the app and the notebook

?


VIRGINIA