YELLOWBRICK: STEERING SCIKIT-LEARN WITH VISUAL TRANSFORMERS

By Benjamin Bengfort and Rebecca Bilbro

DistrictDataLabs

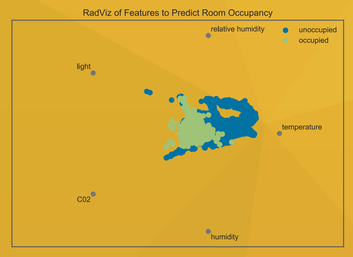
pycon.districtdatalabs.com

PORTLAND, OREGON PYCON 2017

WHICH FEATURES DO I USE?

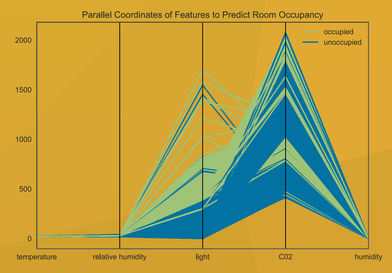
Given labeled data about rooms...

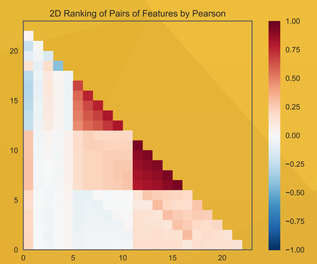
- Which features are most predictive?
- Empty or occupied?



Radviz and ParallelCoordinates

Use Yellowbrick Radial Visualizations or Parallel Coordinates to look for class separability!



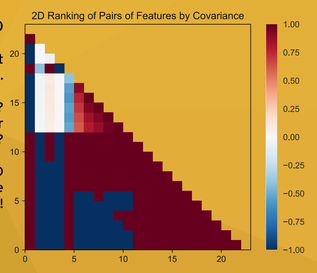


Rank2D

Given labeled data about credit card default...

- Feature relationships?
- Correlations and/or collinearity?

Use Yellowbrick Rank2D for pairwise feature analysis!



THE API: SCIKIT-LEARN

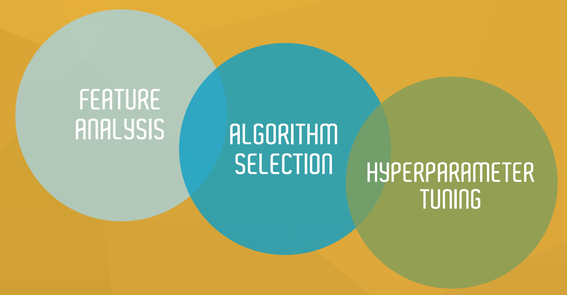
Scikit-Learn has so many models, making automated model selection very convenient!

```
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import AdaBoostClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier

from sklearn import cross_validation as cv

classifiers = [
    KNeighborsClassifier(5),
    SVC(kernel="linear", C=     ),
    RandomForestClassifier(max_depth=5),
    AdaBoostClassifier(),
    GaussianNB(),
]

kfold  = cv.KFold(len(X), n_folds=12)
max([
    cv.cross_val_score(model, X, y, cv=kfold).mean
    for model in classifiers
])
```

Except ...

- search is difficult, high dimensional.
- even with clever optimization, no guaranteed solution.
- time increases exponentially with search space.

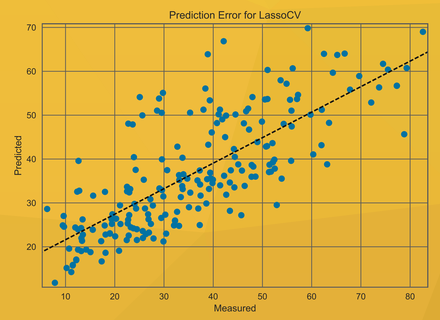
The Model Selection Triple (Arun Kumar, et al):



WHICH MODEL SHOULD I USE?

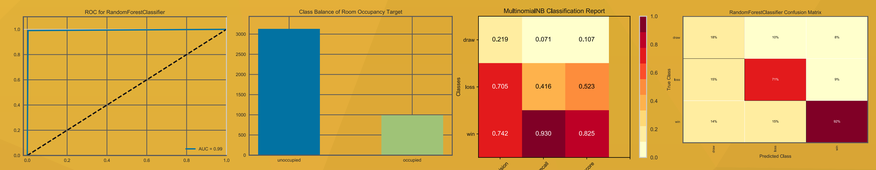
Prediction Error and Residuals Plot

Visualize the distribution of error to diagnose heteroscedasticity:



ROCAUC, Classification Report, Confusion Matrix, and Class Balance

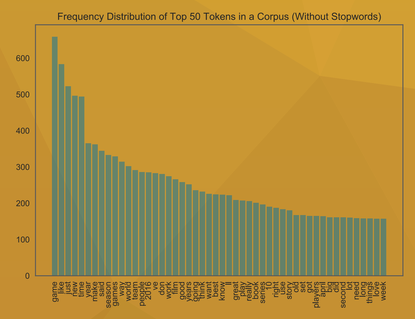
ROCAUC helps us see overall accuracy; classification heatmap helps distinguish Type I & Type II error; and confusion matrix shows us error on a per-class basis. What to do with a low-accuracy classifier? Check for imbalance!



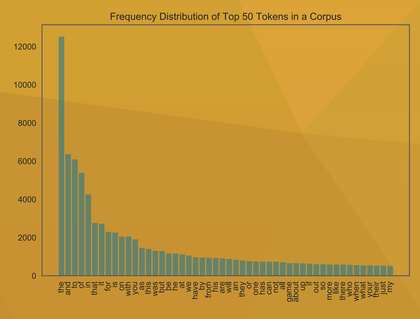
WORKING WITH TEXT DATA

Text data is notoriously high-dimensional and hard to visualize. Yellowbrick can help!

Frequency Distributions

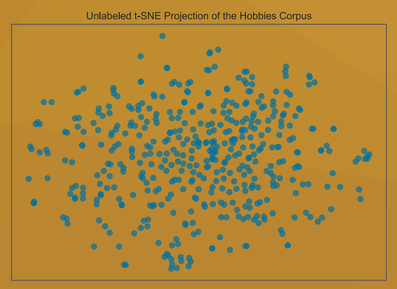


Visualize important word features: before stopwords removal...

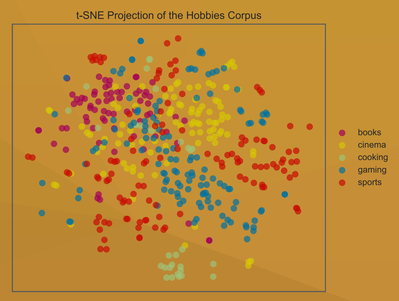


and after!

t-SNE



Visualize the distribution of corpus documents in 2 dimensions



ENTER YELLOWBRICK

scikit-yb.org

Yellowbrick is a new Python library that:

- extends the Scikit-Learn API.
- enhances the model selection process.
- provides visual tools for feature analysis, diagnostics & steering.

In Scikit-Learn:

```
# Import the estimator
from sklearn.linear_model import Lasso

# Instantiate the estimator
model = Lasso()

# Fit the model to the estimator
model.fit(X_train, y_train)

# Generate a prediction
model.predict(X_test)
```

With Yellowbrick:

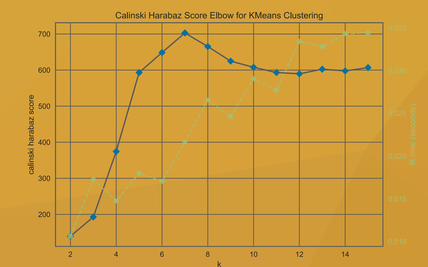
```
# Import the model and visualizer
from sklearn.linear_model import Lasso
from yellowbrick.regressor import PredictionError

# Instantiate the visualizer
visualizer = PredictionError(Lasso())

# Fit
visualizer.fit(X_train, y_train)

# Score and visualize
visualizer.score(X_test, y_test)
visualizer.poof()
```

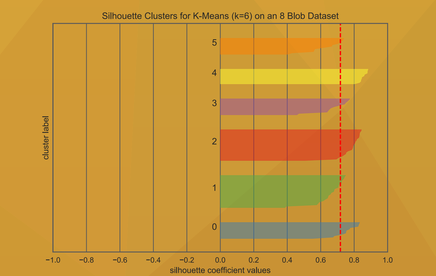
HOW DO I TUNE MY MODEL?

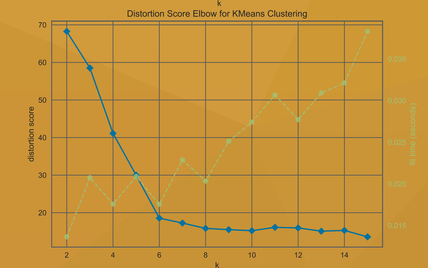
Elbow Curves and Silhouette Scores

- How do you pick an initial value for k in k-means clustering?
- How do you know whether to increase or decrease k?
- Is partitive clustering the right choice?

Higher silhouette scores mean denser, more separate clusters:







Alpha Selection

- Should I use Lasso, Ridge, or ElasticNet?
- Is regularization even working?