Fit 2 Stitch
Esther Liu, Arthur Qin, Sakshi Singla

Predicting the fit of a clothing item

Overview

- 1. Ask: Our Problem
- 2. Acquire: Our Dataset
- 3. Process: Feature Engineering
- 4. Model: Model Comparison
- 5. Deliver: Conclusion

1. Ask: Our Problem

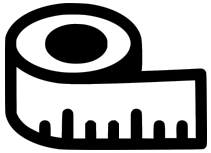
“Fast fashion is the second most polluting industry in the world.”

2. Acquire: Our Dataset

- 192,544 rows, 28 col

RENT THE RUNWAY

Fit Feedback

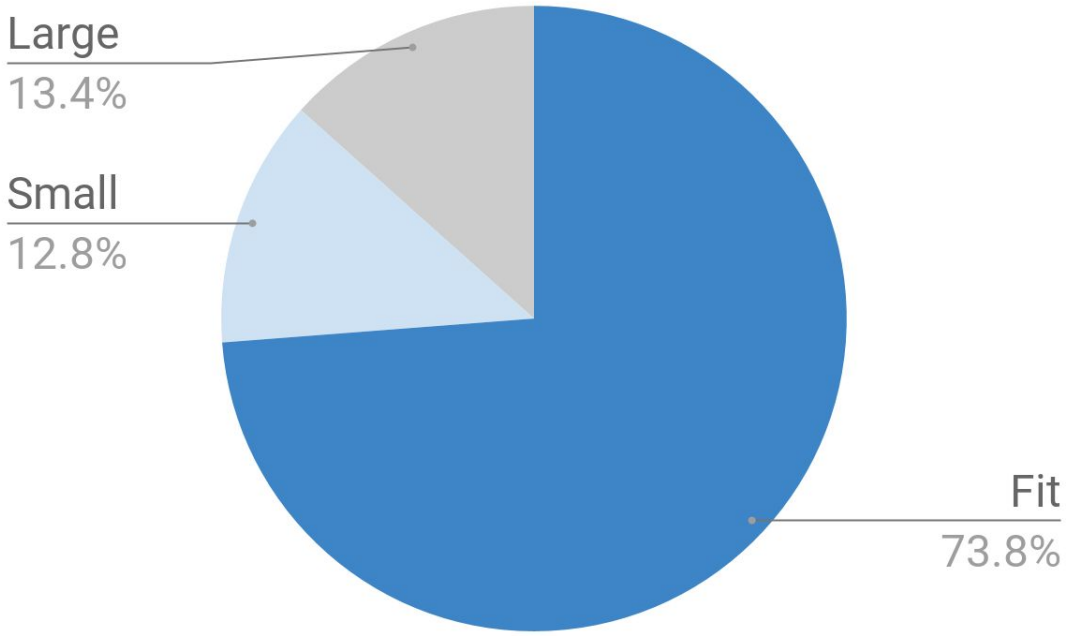


Customer Measurement

Clothing Item

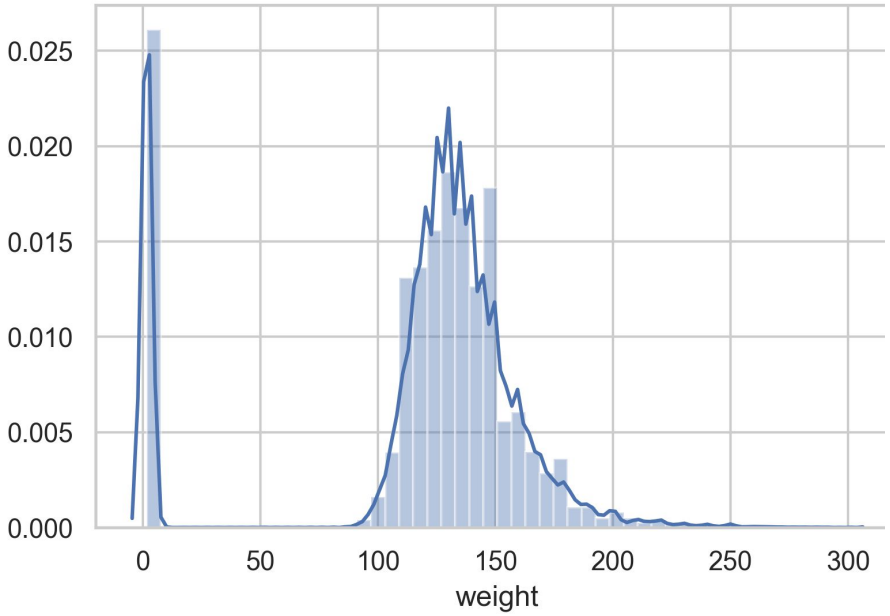
Ratings & Reviews

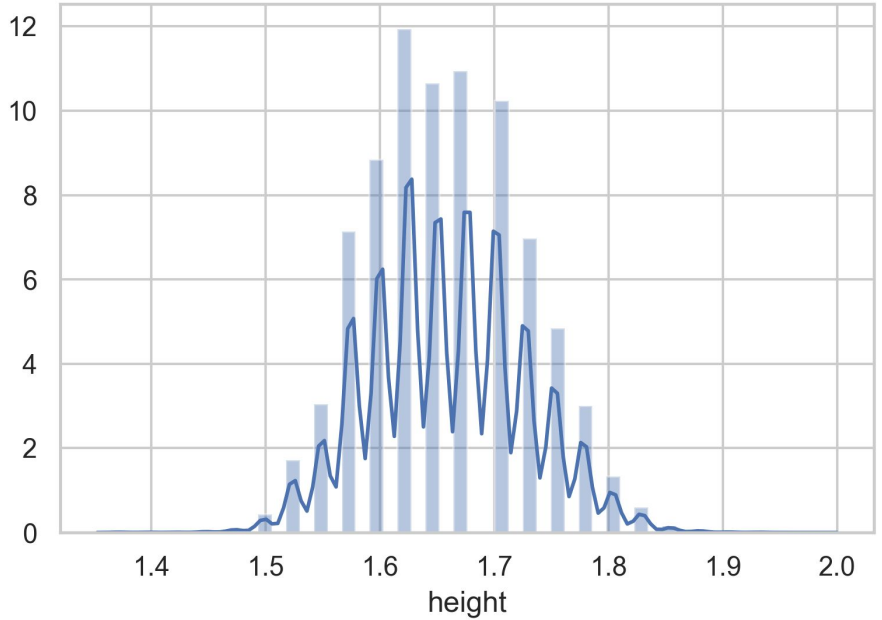
Fit Feedback



Customer Measurement

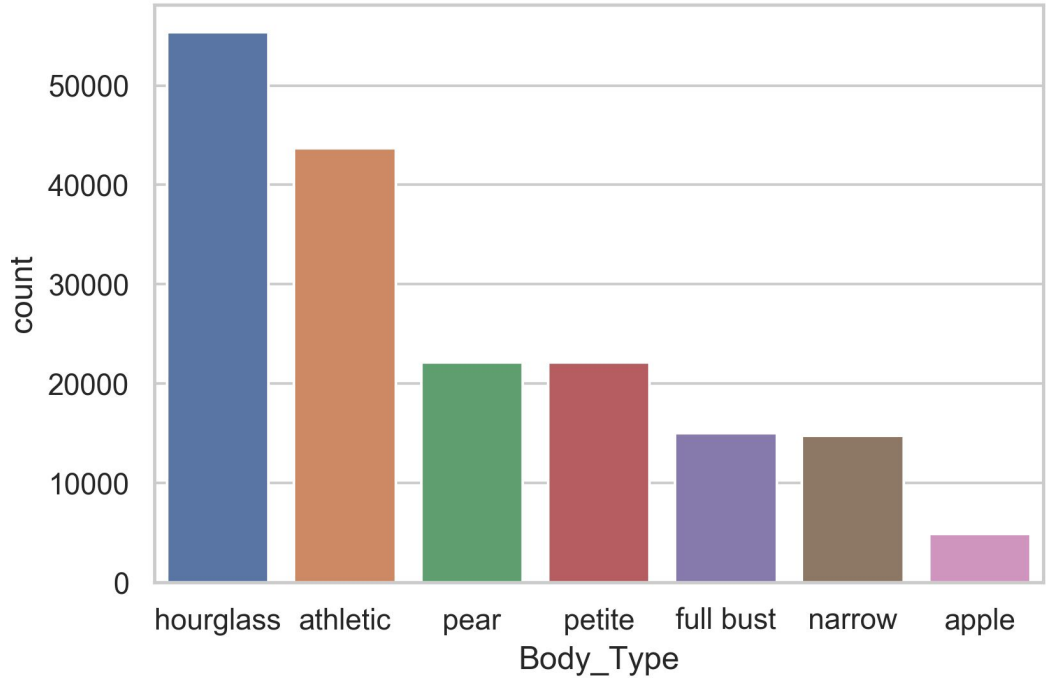
Weight, Height, Bust Size, Body Type





Customer Measurement

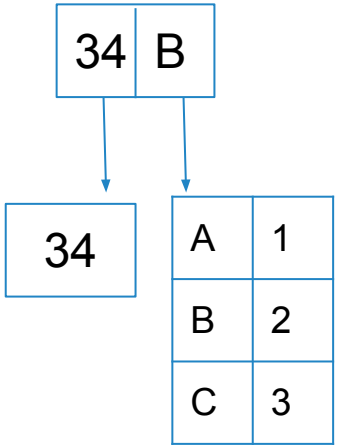
Weight, Height, Bust Size, Body Type



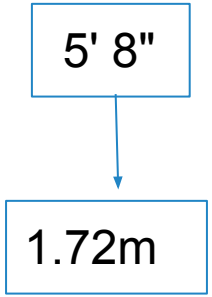
3.
Process:
Feature Engineering

Categorical → Numeric

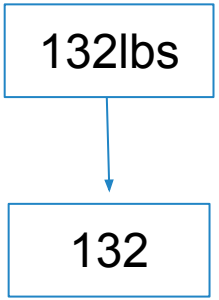
Bust Size



Height



Weight



Frequency Encoding

	item_id	items_count
0	2803	28
1	1196	519
2	145	81
3	563	2241
4	5082	114

	body_type	body_type_count
0	hourglass	55349.0
1	straight & narrow	14742.0
3	pear	22135.0
4	athletic	43667.0
5	athletic	43667.0

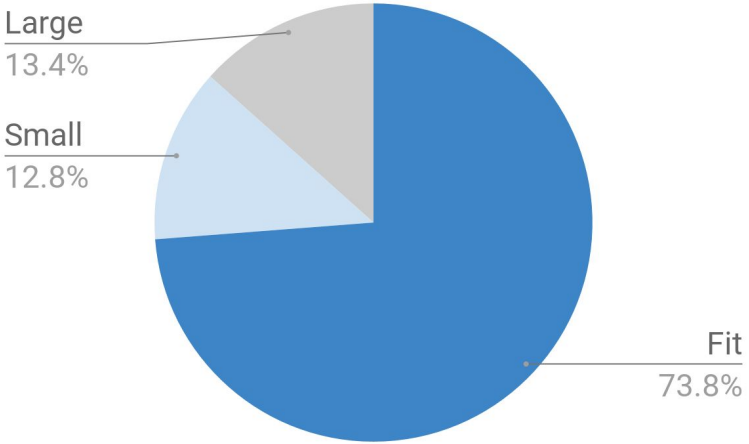
Label Encoding

→ **Each category becomes a unique numeric value**

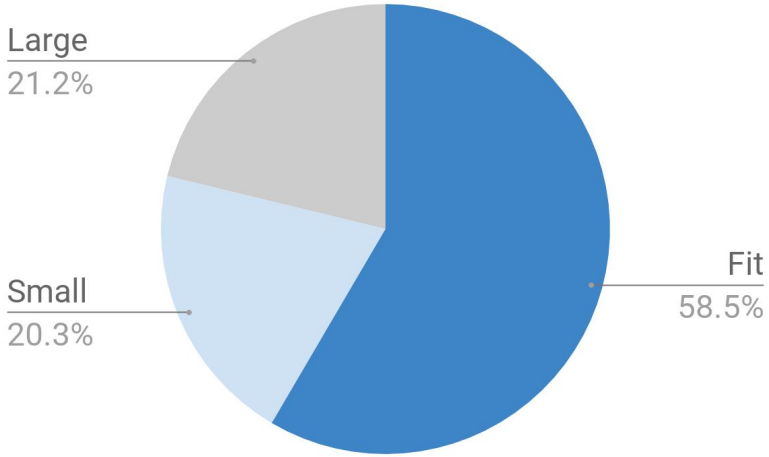
- Many categorical levels
- Works well with RF model

Unbalanced fit value

- Proper train-test split
- Upsample *Small* and *Large* x 2



Before upsampling



After upsampling

4.
Our Model

Our pipeline

height, size, weight, bust size modified, body type count,
rating, body type, category, item id, duration rented for, age range

Preprocessor

Most of our features are numeric now

Simple Imputer with strategy='median'

Standard Scaler

Applied to k-nearest neighbors (KNN) and Logistic Regression

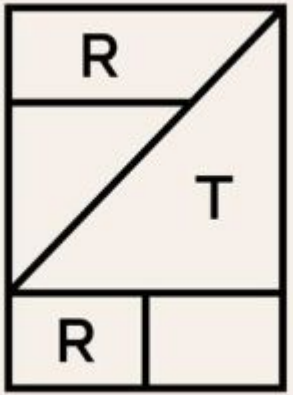
Classifier

RandomForestClassifier
GaussianNB
KNeighborsClassifier
LogisticRegression

Fine tune our hyperparameters through random search with cross validation

bootstrap=True
criterion='gini'
max_features=8
min_samples_leaf=1
min_samples_split=4
n_estimators=100

Our F1 score increased from 0.6653 to 0.6703
Accuracy score increased from 0.7202 to 0.7224



Why do we choose F1 score as our North Star Metric?

F1 is more useful than accuracy when we have an uneven class distribution.

Based on the prediction results from our models, we decide that Random Forest perform the best among the four models

K Nearest Neighbors

Accuracy Score
0.5913
F1 Score
0.6026

Logistic Regression

Accuracy Score
0.7269
F1 Score
0.6647

Naive Bayes

Accuracy Score
0.7310
F1 Score
0.6673

Random Forest

Accuracy Score
0.7224
F1 Score
0.6703

Random Forest Feature Importances

Size, rating, and weight seems to be the most important features

Conclusion

Deliver

Size

Obviously the size of an item matters!

Rating

If it fits others, it might fit you too!

Weight

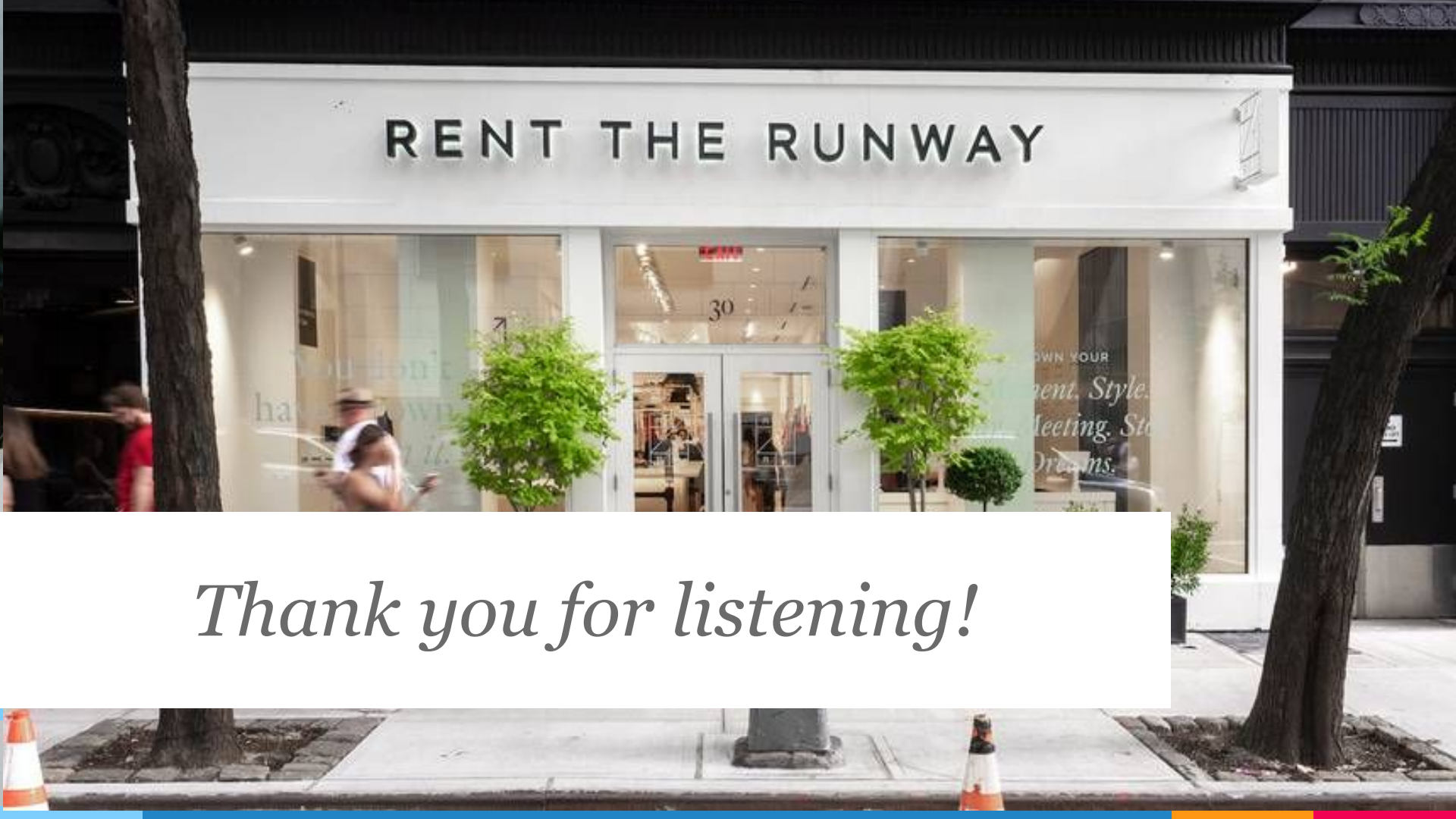
Well, no comment on this (smiley face)

Summary and Business Implications

- Random forest performs the best
- We could use size of the item, rating of the item, and a customer's weight to provide item recommendation
- Better fit helps us reduce shipping and returning costs

Limitations and potential areas for improvement (if we have more time...)

- *Collinearity and codependency across features*
- *Other methods of getting feature importance*
- *Better handling missing values by creating an boolean indicator column*



Thank you for listening!