Practical Machine Learning in R

Feature Engineering

Lars Kotthoff[1,2]
larsko@uwyo.edu

[1] with slides from Bernd Bischl and Michel Lang
[2] slides available at http://www.cs.uwyo.edu/~larsko/ml-fac

Why do we care?

- reduce dimensionality
- increase interpretability
- increase predictive performance

Feature Selection

Filter Preliminary step, independent of model

Embedded Learner has feature selection embedded, e.g. random forests

Wrapper iteratively and transparently find best features for particular learner

Feature Filters

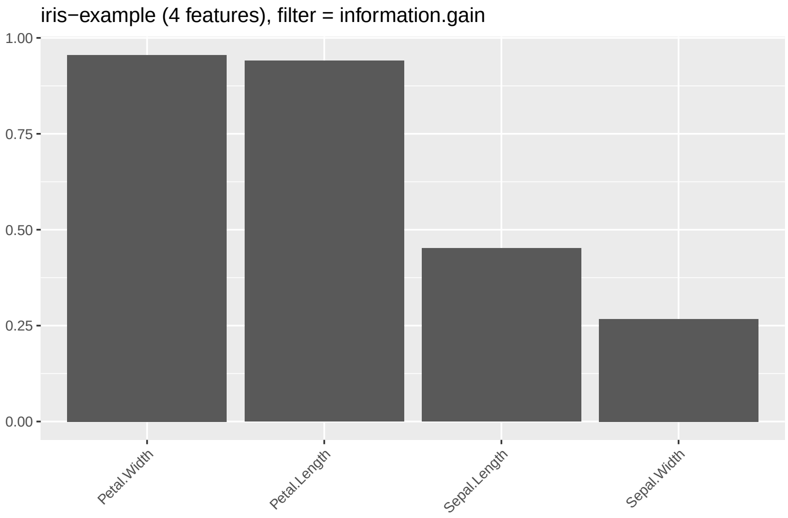
- Numerical score that measures influence on prediction
- Often independent of learner
- Often fast to compute
- Can be used to rank features and select based on threshold
- Can be misleading

Filter Examples

- Correlation between feature and target in regression
- Mutual information between feature and target in classification

```
## FilterValues:
## Task: iris-example
##           name    type information.gain
## 1 Sepal.Length numeric        0.4521286
## 2  Sepal.Width numeric        0.2672750
## 3 Petal.Length numeric        0.9402853
## 4  Petal.Width numeric        0.9554360
```

Filter Examples



Embedded Feature Selection

- model-specific measure of feature importance
- requires support from learner implementation
- most useful for post-hoc feature analysis

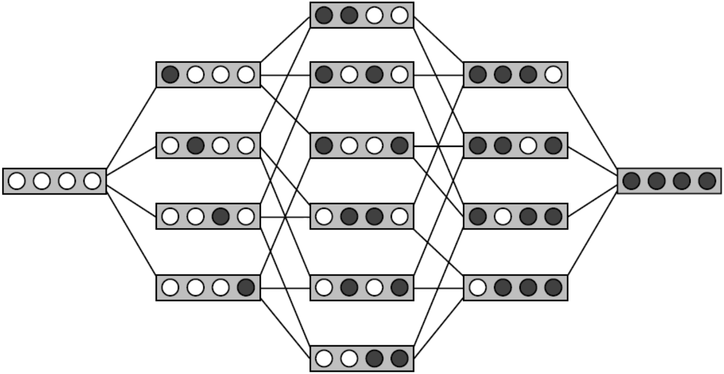
Embedded Feature Selection

```
## FeatureImportance:
## Task: iris-example
##
## Learner: classif.randomForest
## Measure: NA
## Contrast: NA
## Aggregation: function (x)  x
## Replace: NA
## Number of Monte-Carlo iterations: NA
## Local: FALSE
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     9.730877    2.530707     42.76649    44.22676
```

Feature Selection Wrapper

- Evaluate feature sets with learner, e.g. by cross-validation
- Measures probably what you are interested in
- Will be slow in very high-dimensional spaces
- Several methods available

Feature Selection Wrapper



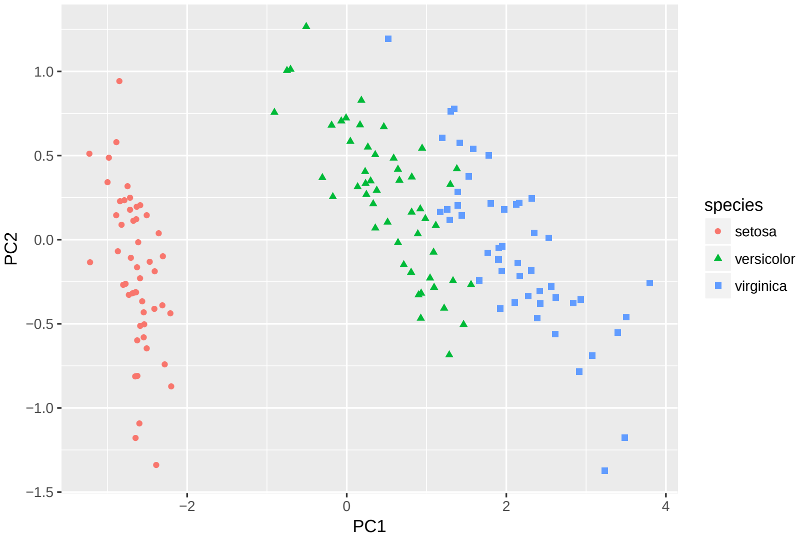
Feature Selection Wrapper

```
## Features          : 1
## Performance       : mmce.test.mean=0.0467
## Petal.Width
##
## Path to optimum:
## - Features:    0  Init    :                      Perf = 0.7  Diff: NA  *
## - Features:    1  Add     : Petal.Width          Perf = 0.046667  Diff: 0.65333  *
##
## Stopped, because no improving feature was found.
```

Principal Component Analysis

- project into lower-dimensional feature space
- dimensions are the uncorrelated principal components
- principal components are combinations of original features that account for variation
- first principal component accounts for most of the variance in the data
- helpful in visualization
- `http://setosa.io/ev/principal-component-analysis/`

Principal Component Analysis



Feature Expansion

- add combinations of features (e.g. products) as new features
- consider pairs, triples... of features
- can allow linear model to learn non-linear relationships
- usually not necessary for complex models

Exercises

`http://www.cs.uwyo.edu/~larsko/ml-fac/06-features-exercises.Rmd`