2022 MathWorks
中国汽车年会

人工智能算法的工程应用

——以驾驶风格识别和速度预测为例

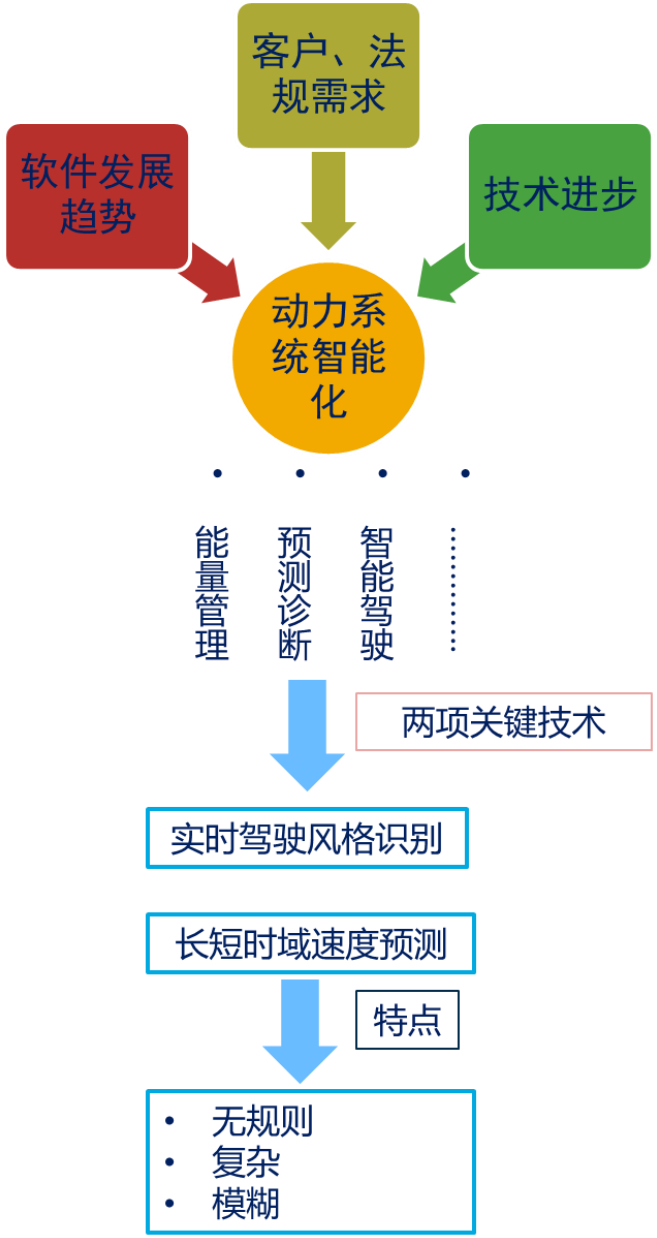
陈松波, 广汽研究院

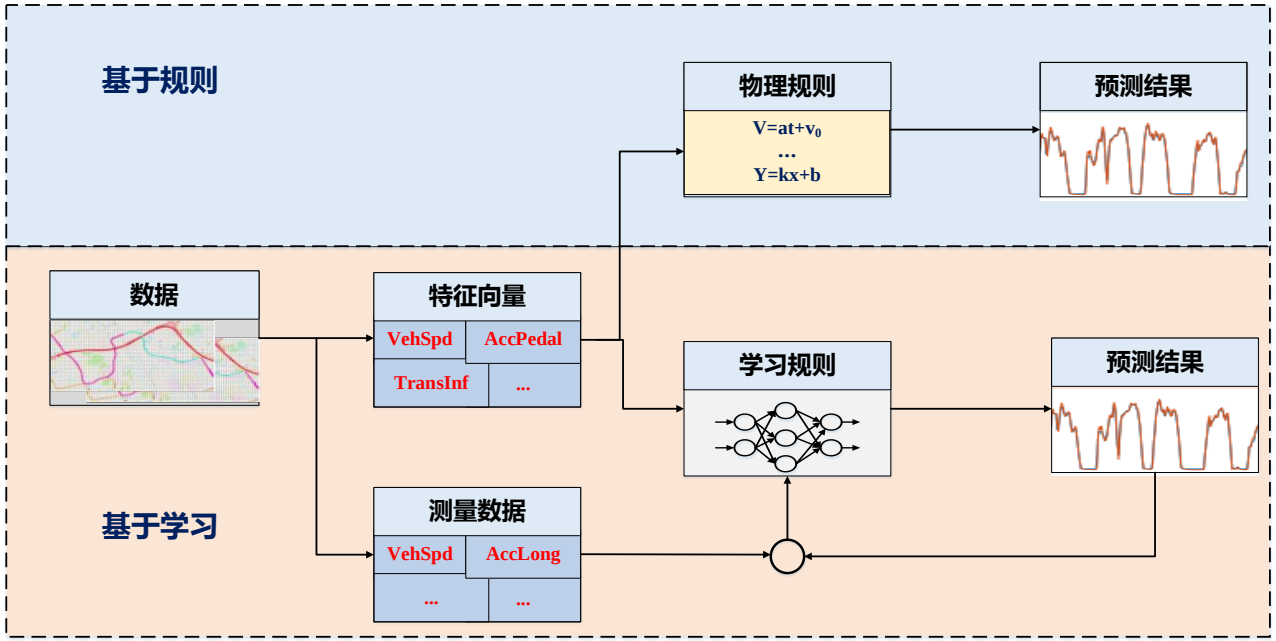
MathWorks

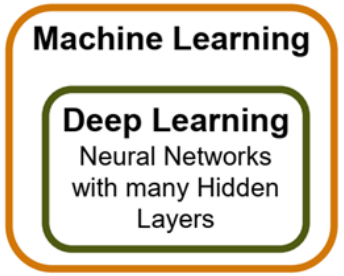
主要内容

- 项目背景
- 动力系统智能化软件开发
  - 基于机器学习的驾驶风格识别算法开发
  - 基于深度学习的速度预测算法开发
- 总结

项目背景


客户、法规需求
软件发展趋势
技术进步
动力系统智能化
能量管理
预测诊断
智能驾驶
……
两项关键技术
实时驾驶风格识别
长短时域速度预测
特点
无规则
复杂
模糊

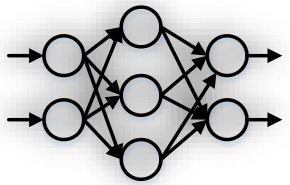

基于规则
物理规则
V=at+v0
…
Y=kx+b
预测结果
数据
特征向量
VehSpd
AccPedal
TransInf
…
学习规则
预测结果
基于学习
测量数据
VehSpd
AccLong
…
…


Machine Learning
Deep Learning
Neural Networks with many Hidden Layers

- 数据挖掘
- 预测

为什么选择MATLAB?

数据处理和可视化



AI算法开发

Simulink

MBD模型化开发

Documentation

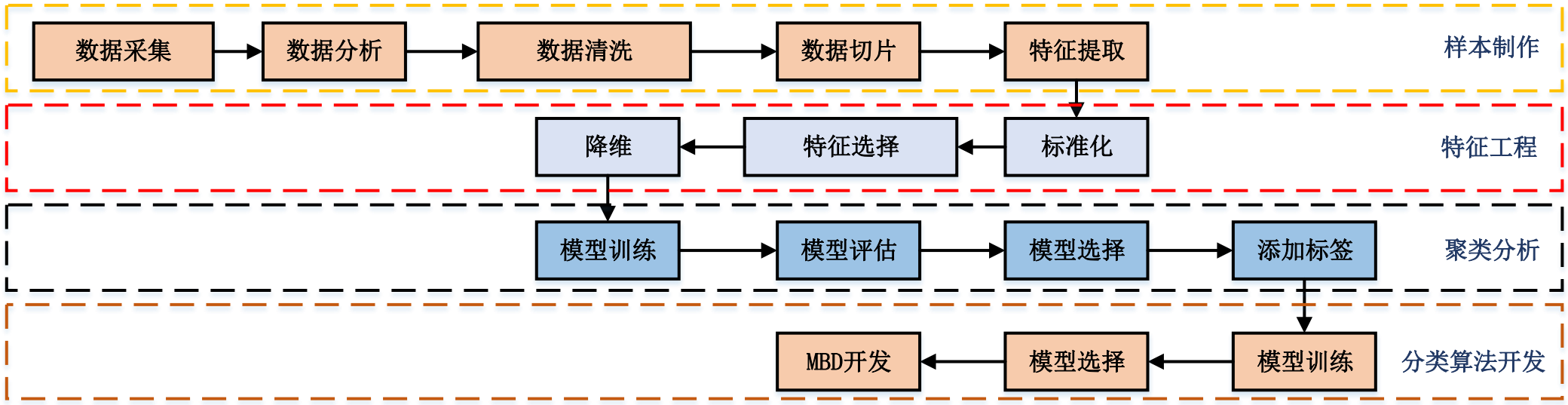
帮助文档

驾驶风格识别算法开发

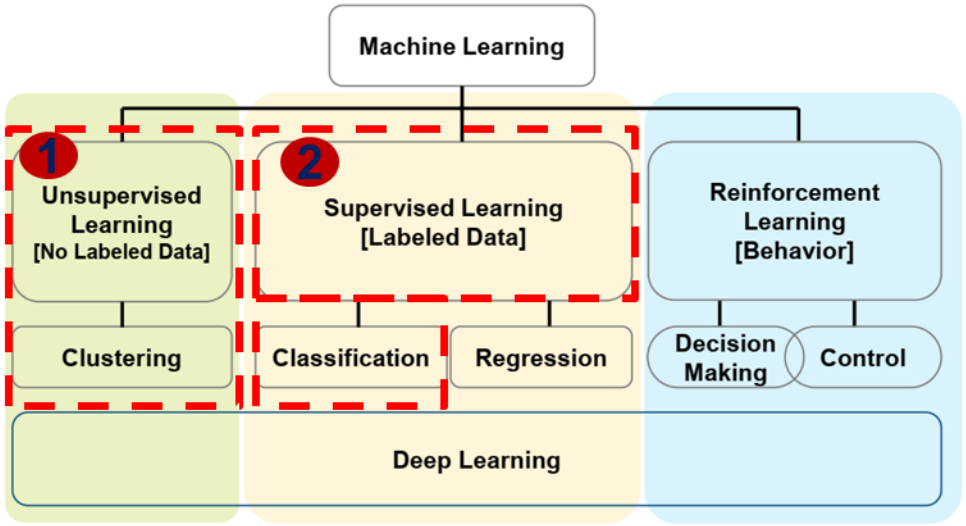
开发简介

- 经济性；
- 动力性；
- 智能化软件开发；

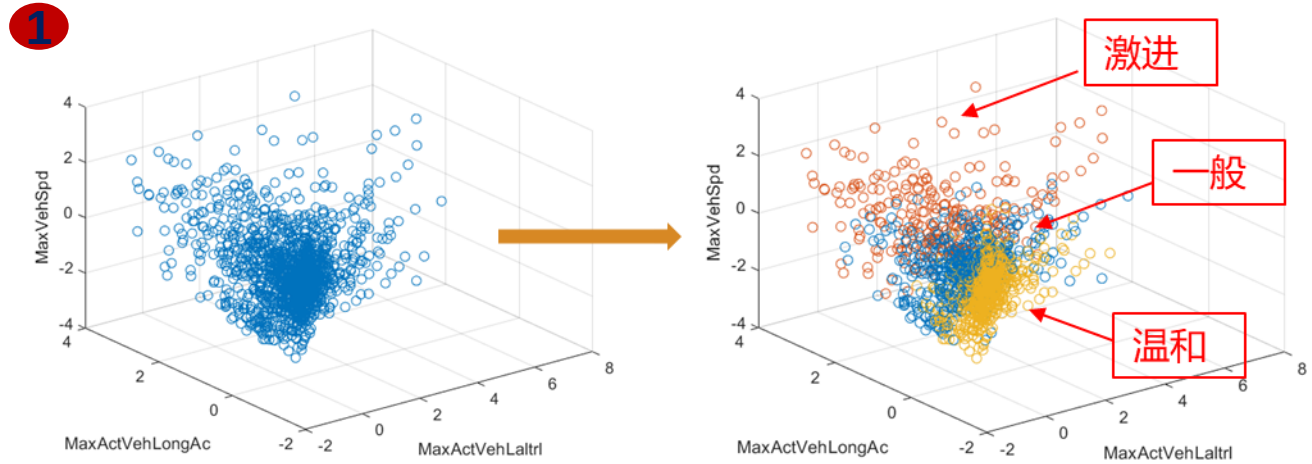
技术路线



技术点



1



2

分类算法训练(SVM/KNN/分类树)

最大车速

平均车速

最大横向加速度

⋮

最大加速踏板变化率

分类算法

驾驶风格标签

- 激进
- 一般
- 温和

具体实现：

数据采集

- 驾驶循环的制定
- 驾驶员的选择

数据分类

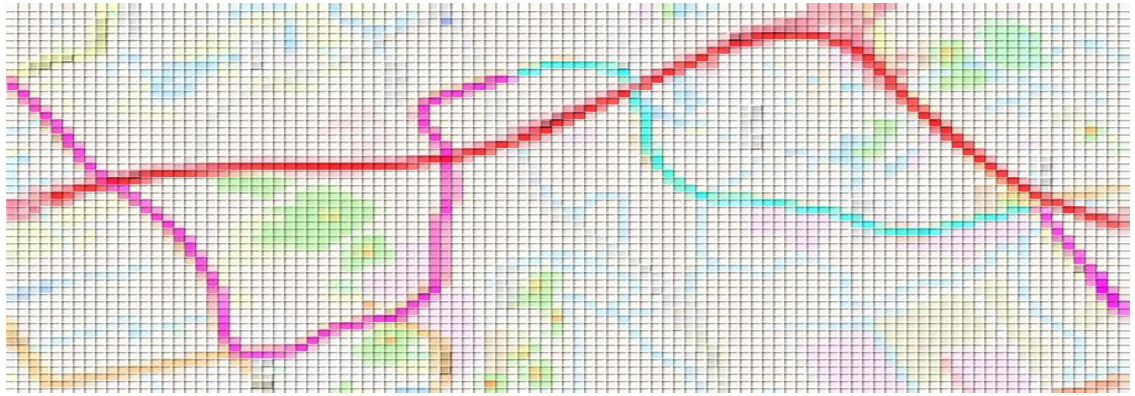
- 道路类型分类
- 驾驶行为分类

特征提取和标准化

- 数学特征提取
- 标准化处理

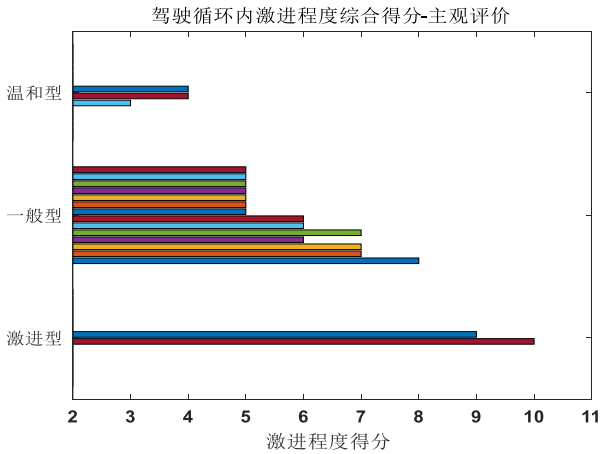
特征选择

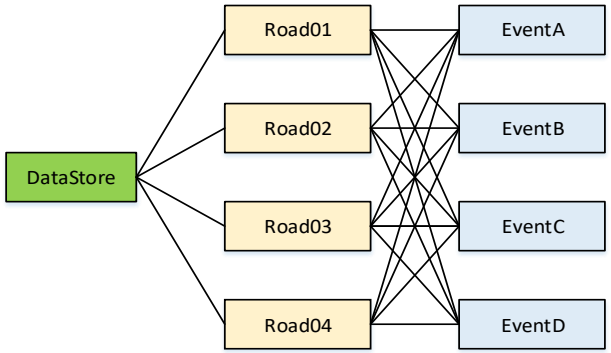
- 方差分析
- 双样本t检验
- 相关性分析

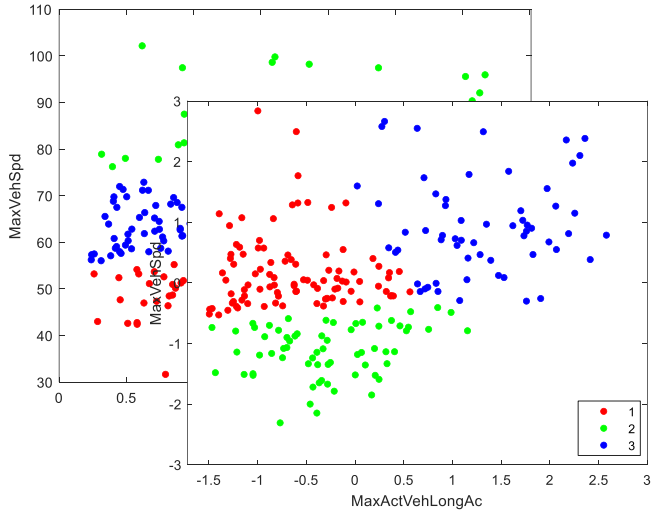


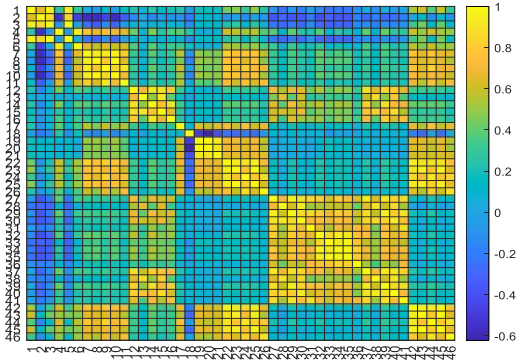
```
%取最大值
VarMaxFea=max(table2array(FileData.EventSampleData{NumSamp,1}),[],1);
%取最小值
VarMinFea=min(table2array(FileData.EventSampleData{NumSamp,1}),[],1);
%取平均值
VarMeanFea=mean(table2array(FileData.EventSampleData{NumSamp,1}),1);
%取方差
VarVarFea=var(table2array(FileData.EventSampleData{NumSamp,1}),1);
%取标准差
VarStdFea=std(table2array(FileData.EventSampleData{NumSamp,1}),1);
%取均方根
VarRmsFea=rms(table2array(FileData.EventSampleData{NumSamp,1}),1);
```

$$R_{cor}=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}}$$







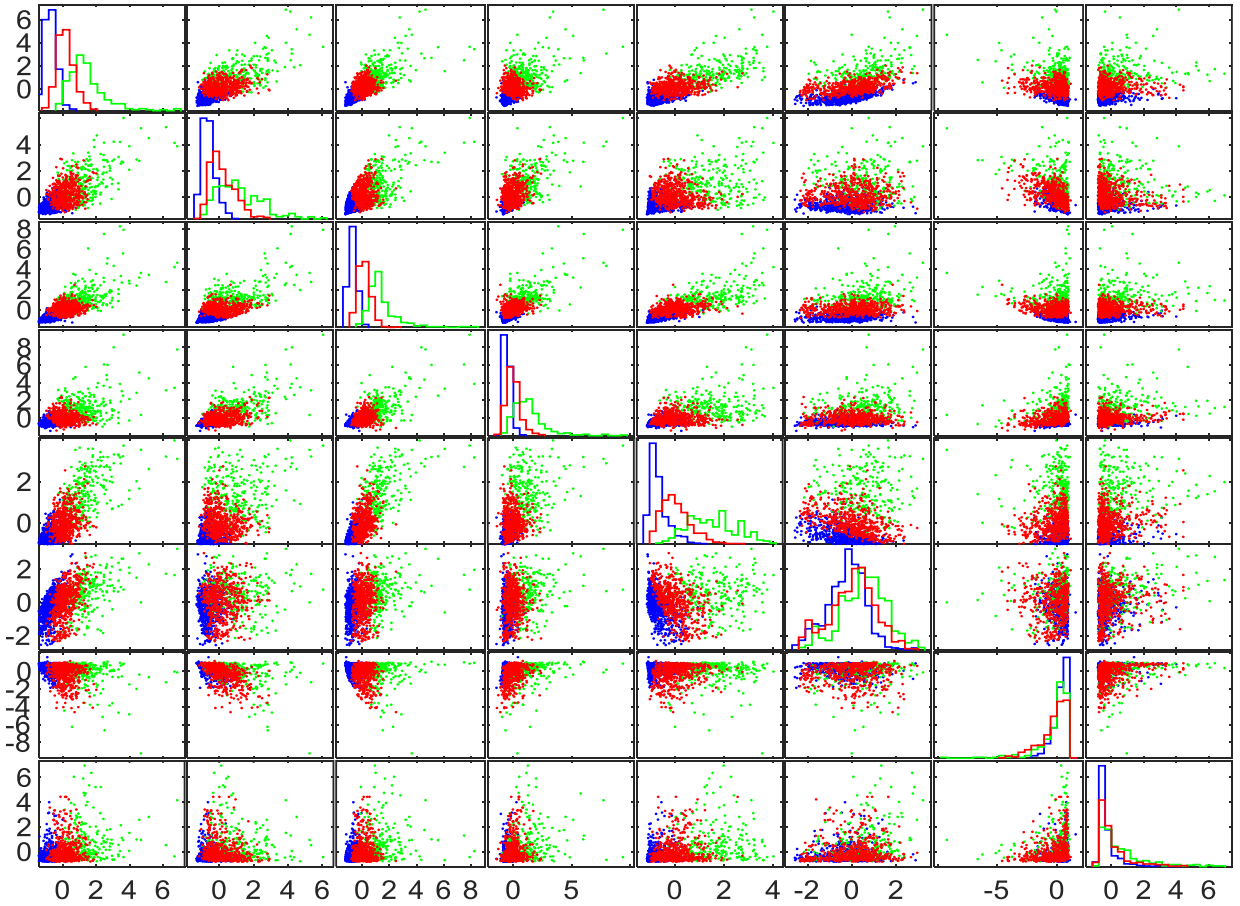


聚类分析

常见聚类算法				
Kmeans	GMM	层次聚类	DBSCAN密度聚类	Spectral Clustering谱聚类

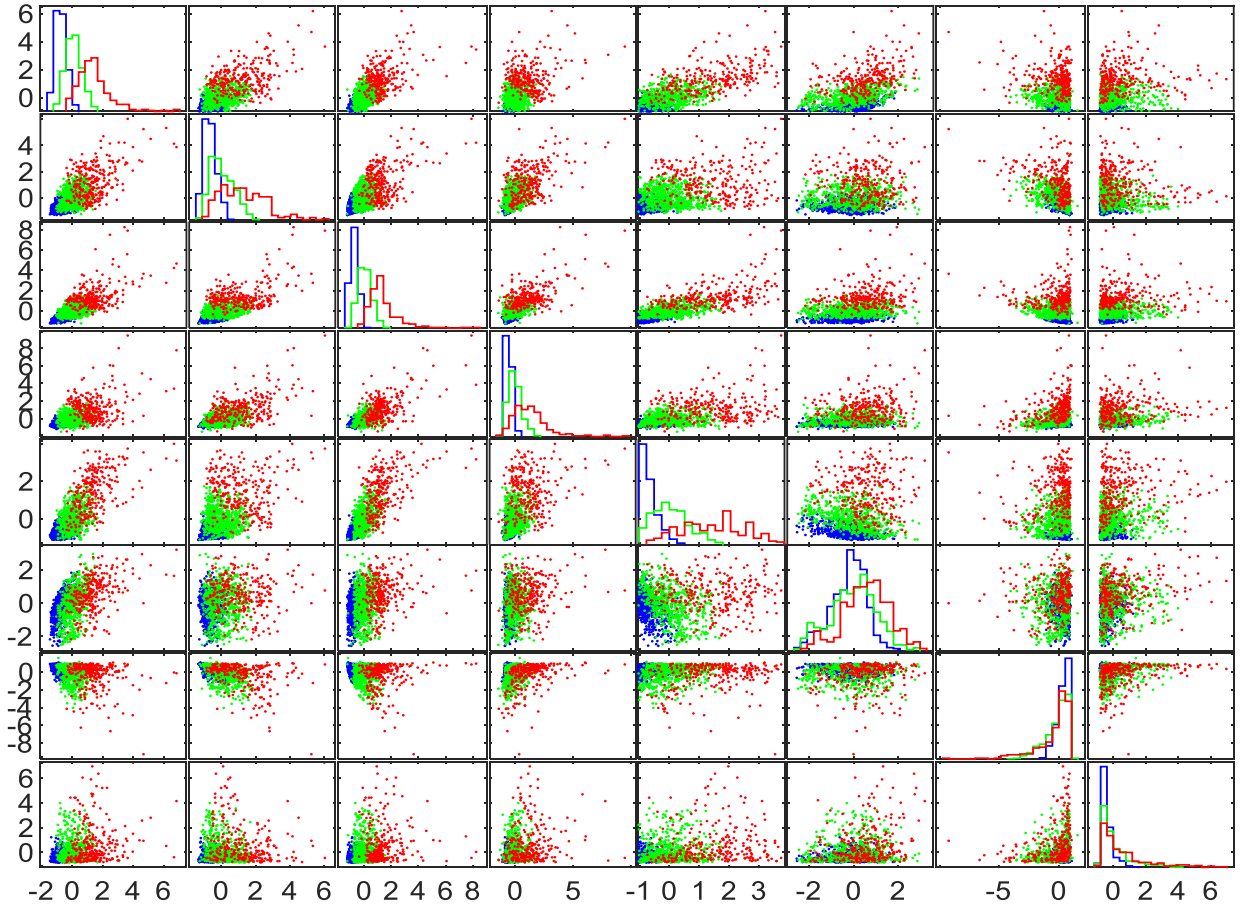
- Kmeans聚类_无PCA

```
KData=cast(TestSampleSet{:,2:end},'single');
%设置模型训练参数
opts = statset('Display','off','MaxIter',10000000,...
'TolFun',1e-6,'UseParallel',false);
%使用kmeans聚类
TestSampleKmeansIdx = kmeans(KData,3,'distance',...
    'cityblock','Options',opts,...
    'Replicates',1000);
%可视化
gplotmatrix(KData,[],TestSampleKmeansIdx)
```



- 高斯混合模型聚类

```
GmmData=KData;
%训练高斯混合模型
GMModel = fitgmdist(GmmData,3,'Options',opts,...
    'Replicates',1000,'CovarianceType','diagonal',...
    'RegularizationValue',...
    0.01,'Start','plus');
%使用训练好的模型聚类
TestSampleGMMIdx=cluster(GMModel,GmmData);
%可视化
gplotmatrix(KData,[],TestSampleGMMIdx)
```

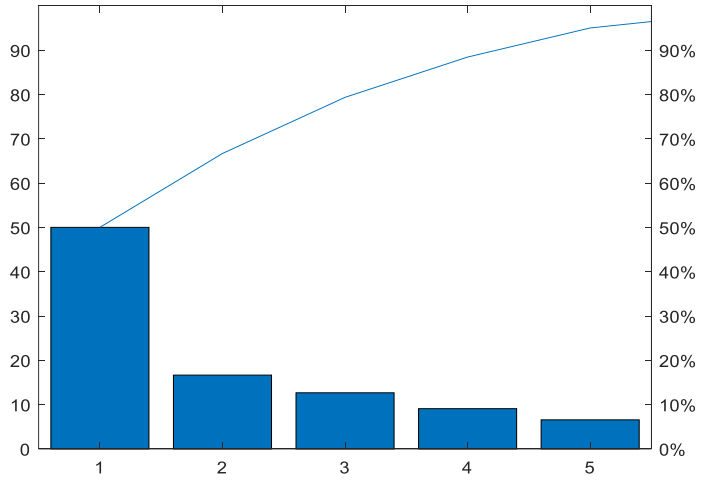


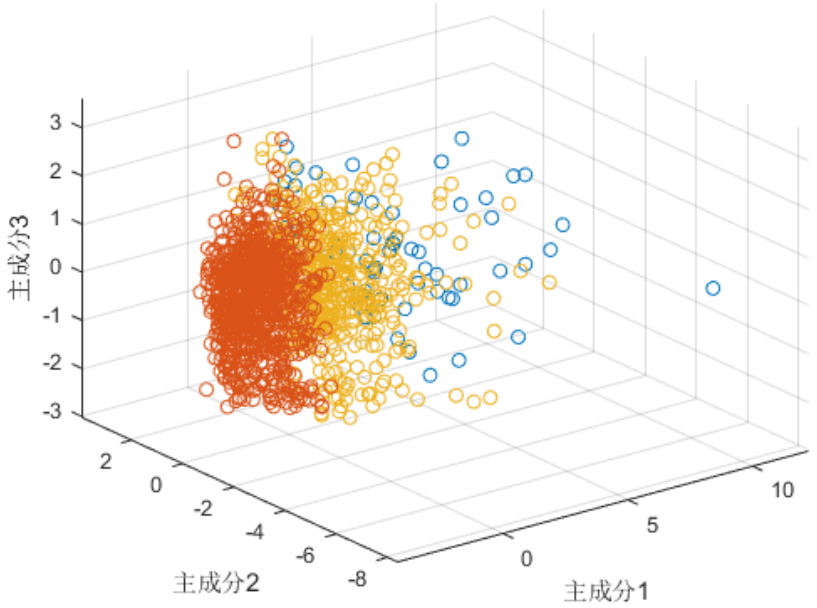
- PCA降维后聚类

```
%PCA降维
[coeff,score,latent,tsquared,explained,mu] = ...
    pca(TestSampleSet{:,2:end});
figure
pareto(explained)
%选取主成分
idx = find(cumsum(explained)>94,1);
featuresTrain = score(:,1:idx);
%Kmeans/GMM聚类
KData=featuresTrain;
GmmData=KData;
opts = statset('Display','off','MaxIter',...
    10000000,'TolFun',1e-6,'UseParallel',false);
TestSamplePCAKmeansIdx = kmeans(KData,3,...
    'distance','sqeuclidean','Options',opts,...
    'Replicates',1000);
GMModel = fitgmdist(GmmData,3,'Options',opts,...
    'Replicates',1000,'CovarianceType','diagonal',...
    'RegularizationValue',0.01,'Start','plus');
TestSamplePCAGMMIdx=cluster(GMModel,GmmData);
```

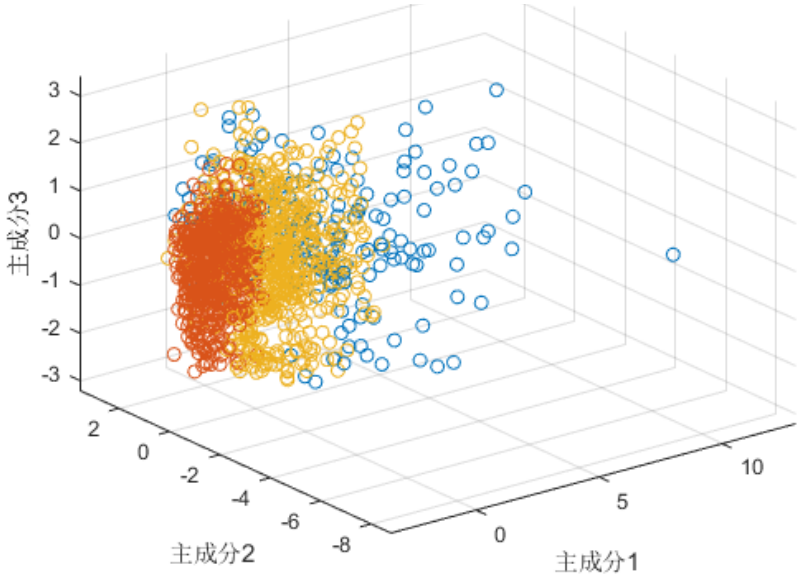
成分1-3：80%

成分	成分方差	成分方差占总方差百分比	成分累计方差
1	3.973895788	49.67370605	49.67370605
2	1.57612133	19.70152092	69.37522888
3	1.067700624	13.34626102	82.72148895
4	0.655951738	8.199398994	90.92089081
5	0.301321626	3.766520977	94.68740845
6	0.196062952	2.450787306	97.13819885
7	0.161447793	2.018097878	99.15629578
8	0.067496605	0.84370774	100





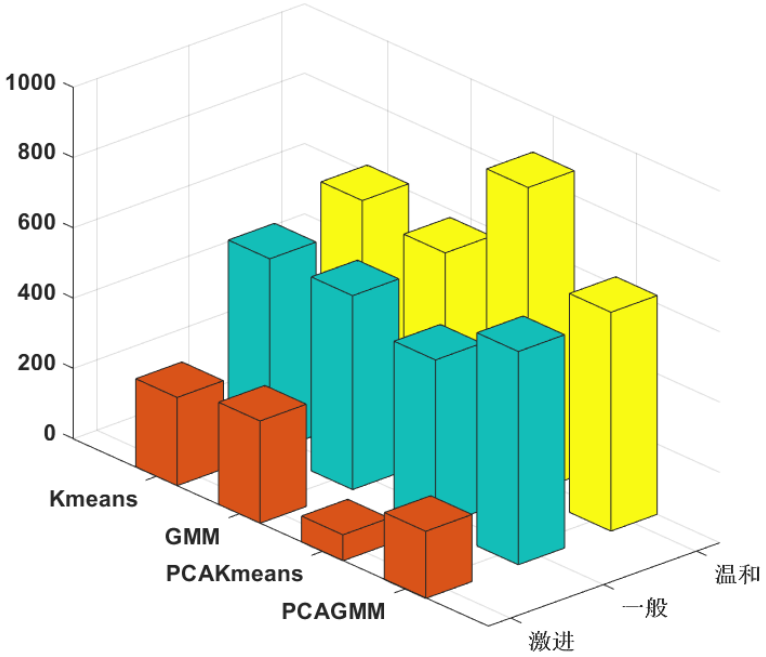
Kmeans



GMM

聚类结果对比

各类型样本数



Kmeans结果

无PCA

	激进(9)	普通(6)	冷静(4)
样本个数	64	74	90
激进型占比	**0.5625**	0.22973	0.055556
一般型占比	0.234375	**0.567568**	0.322222
温和型占比	0.203125	0.202703	**0.622222**

GMM结果

无PCA

	激进(9)	普通(6)	冷静(4)
样本个数	64	74	90
激进型占比	**0.59375**	0.283784	0.055556
一般型占比	0.234375	**0.540541**	0.311111
温和型占比	0.171875	0.175676	**0.633333**

PCA

	激进(9)	普通(6)	冷静(4)
样本个数	64	74	90
激进型占比	**0.359375**	0.054054	0
一般型占比	0.34375	**0.554054**	0.155556
温和型占比	0.296875	0.391892	**0.844444**

PCA

	激进(9)	普通(6)	冷静(4)
样本个数	64	74	90
激进型占比	**0.484375**	0.148649	0.022222
一般型占比	0.34375	**0.648649**	0.233333
温和型占比	0.171875	0.202703	**0.744444**

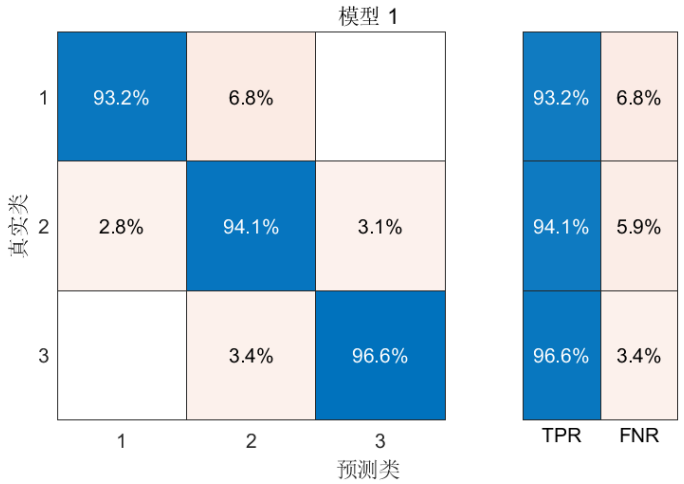
模型训练

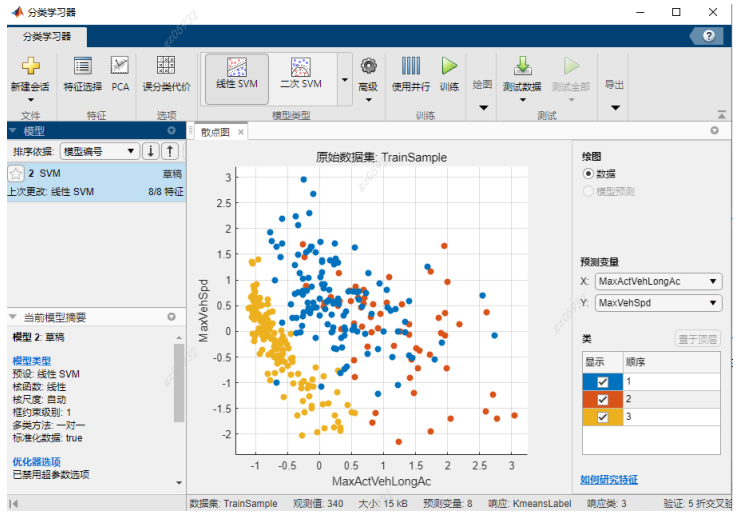
分类算法:

- 支持向量机

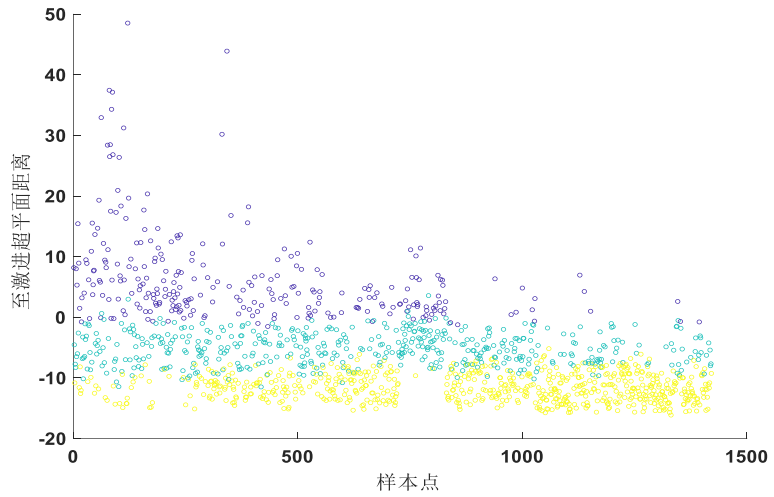
```
%配置训练参数
t = templateSVM('Standardize',false,"KernelFunction"...
    ,"linear","IterationLimit",1e9,...
    "NumPrint",10000,'Solver','SMO',...
    'Verbose',0, 'BoxConstraint',2);
options = statset('UseParallel',true,"Display","Off");
%模型训练
PMdl = fitcecoc(TrainSample(:,1:end-1),...
    TrainSample.KmeansLabel,'Learners',t,...
    "Coding","onevsall","CrossVal","on","KFold",5,...
    "Options",options);
```

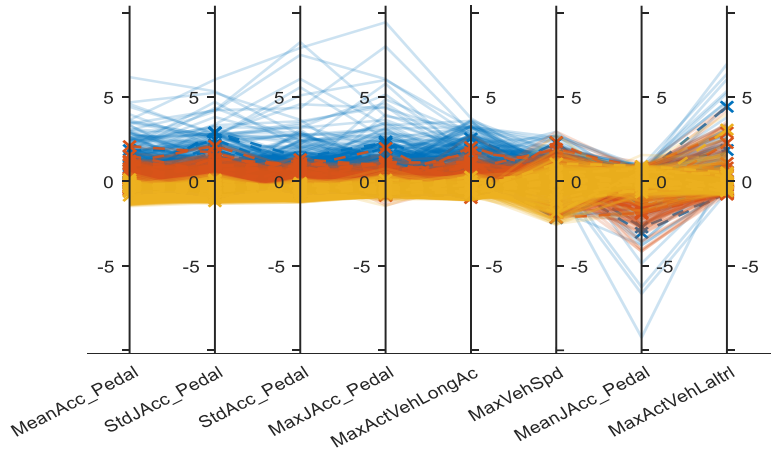
使用脚本训练





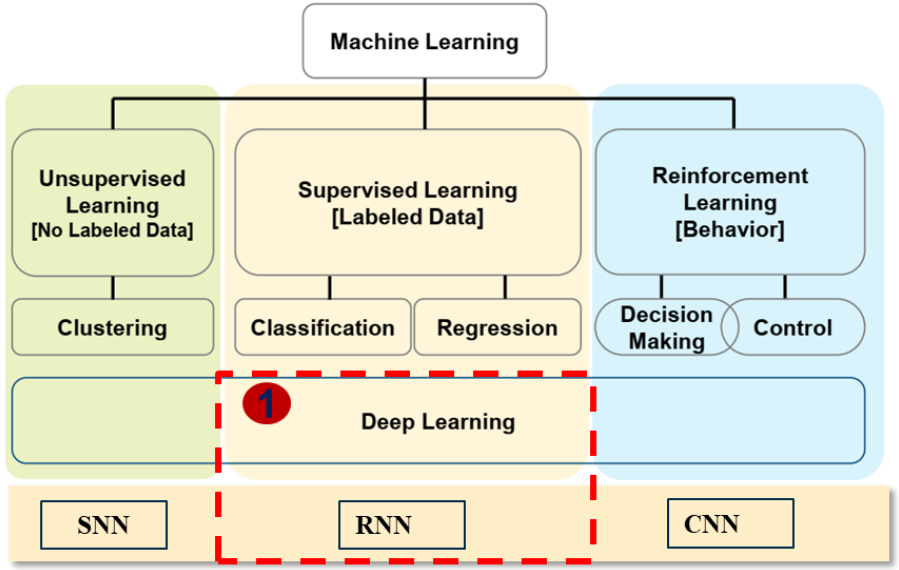
使用分类学习工具箱训练

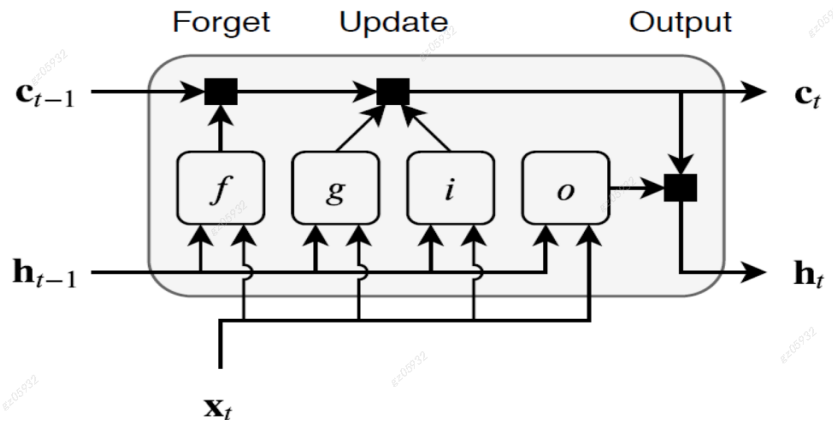




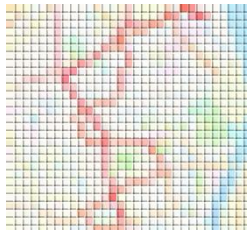
速度预测算法开发

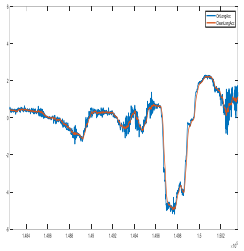
技术点

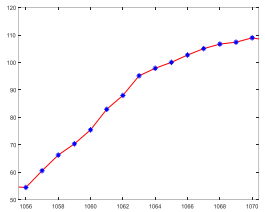


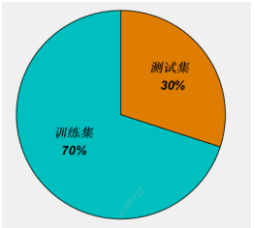


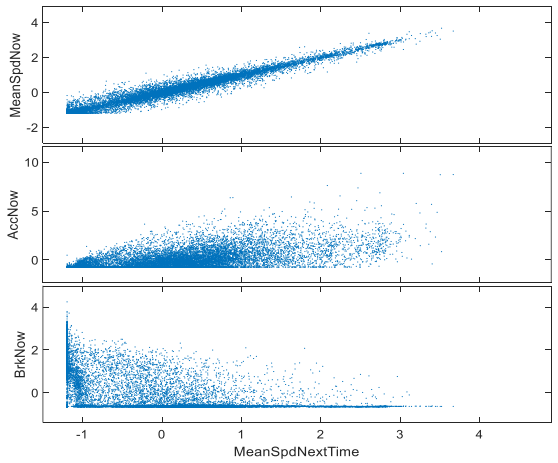
开发流程











Step0.数据获取

↓

Step1.数据清洗

↓

Step2.数据切片

↓

Step3.样本集划分

↓

Step4.特征提取

↓

Step5.标准化

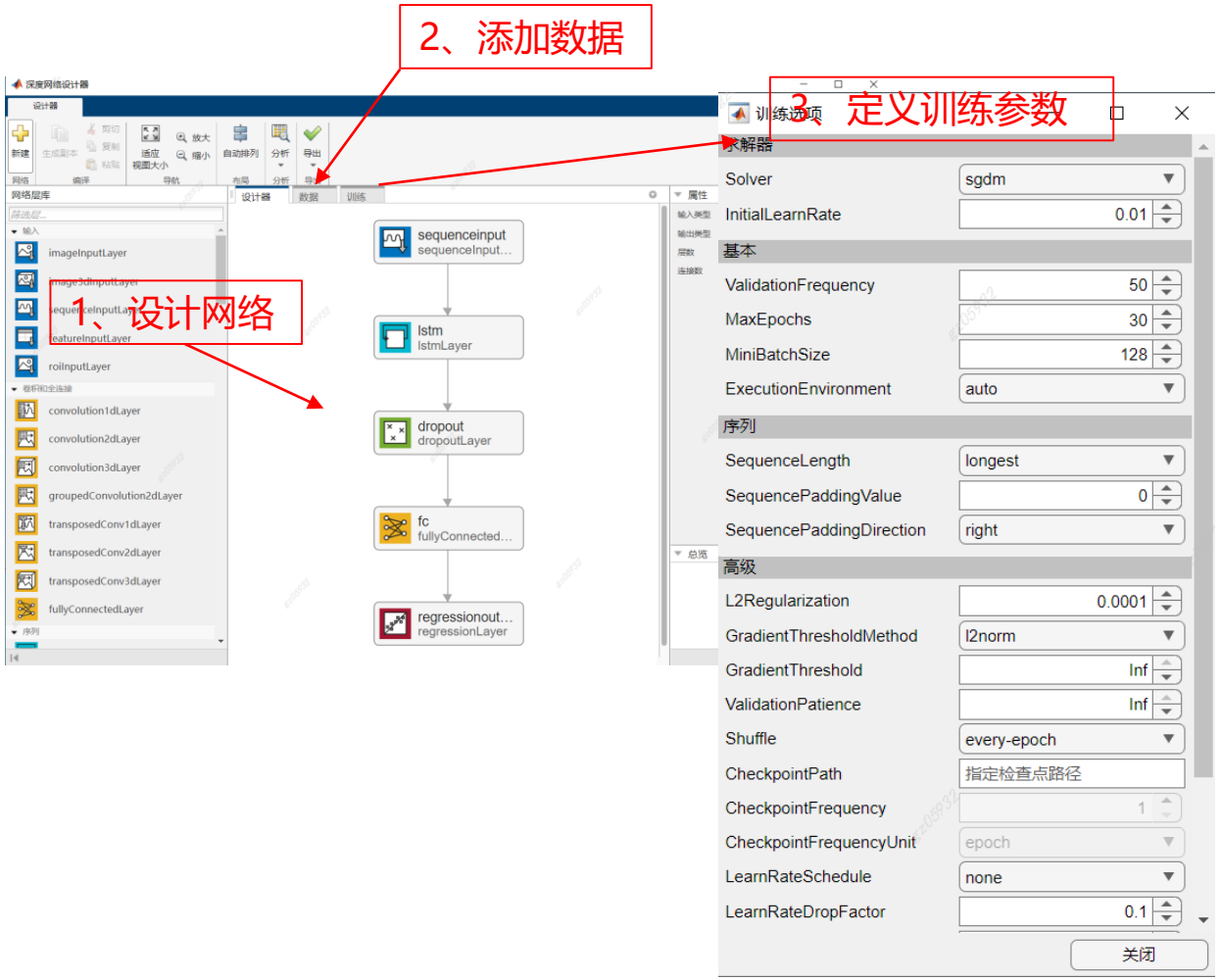
↓

Step6.特征选择

↓

Step7.算法训练

算法开发方式



深度学习工具箱

```
%定义网络结构
layers = [ ...
    sequenceInputLayer(NumFeatures)
    lstmLayer(optVars.numHiddenUnits)
    dropoutLayer(Probability)
    fullyConnectedLayer(numResponses)
    regressionLayer];
%定义训练参数
options = trainingOptions('adam', ...
    "MiniBatchSize",1e10,...
    'MaxEpochs',1000, ...
    'GradientThreshold',1, ...
    'InitialLearnRate',optVars.InitialLearnRate, ...
    'LearnRateSchedule','piecewise', ...
    'LearnRateDropPeriod',500, ...
    'LearnRateDropFactor',0.1, ...
    'L2Regularization',optVars.L2Regularization,...
    "Shuffle","never",...
    'Verbose',0, ...
    'Plots','training-progress',...
    "ValidationData",{XValidation,YValidation},...
    'ValidationFrequency',10,...
    'ValidationPatience',inf);
%网络训练
trainedNet = trainNetwork(XTrain,YTrain,layers,options);
```

脚本定义

- 自定义训练循环

自动微分

- dlarray
- dlgradient

自定义训练循环

- dlnetwork
- forward

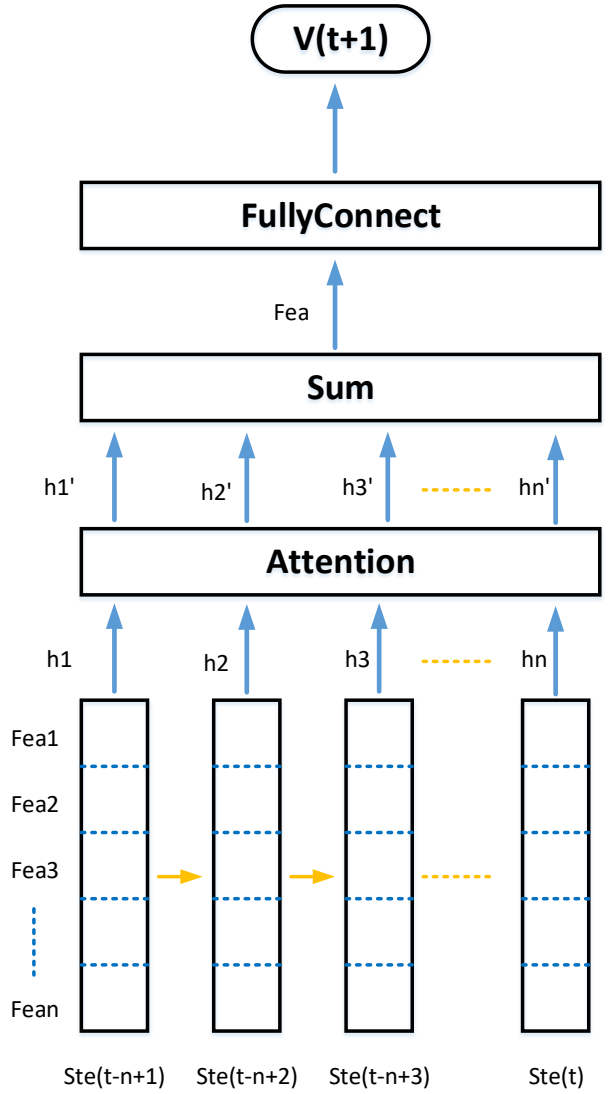
深度学习运算

- dlconv
- dltranspconv
- lstm
- gru
- embed
- fullyconnect
- dlode45
- relu
- leakyrelu
- batchnorm
- crosschannelnorm
- groupnorm

```
iteration = 0;
start = tic;
% 循环迭代.
for epoch = 1:numEpochs
    reset(mbq);
    % 循环一个minibatch.
    while hasdata(mbq)

        iteration = iteration + 1;
        %X: 训练使用的序列样本 T:样本对应标签
        [X,T,sequenceLengthsSource,maskSequenceTarget] = next(mbq);

        % 计算损失和梯度
        [loss,gradients] = dlfeval(@modelLoss,parameters,X,T,sequenceLengthsSource,...
            dropout);

        % Update parameters using adamupdate.   使用adamupdate更新参数
        [parameters,trailingAvg,trailingAvgSq] = adamupdate(parameters,gradients,trailingAvg,trailingAvgSq,...
            iteration,learnRate,gradientDecayFactor,squaredGradientDecayFactor);

        % Display the training progress. Normalize loss by sequence length.
        %  显示训练进度
        D = duration(0,0,toc(start),Format="hh:mm:ss");
        loss = double(loss);
        addpoints(lineLossTrain,iteration,loss)
        title("Epoch: " + epoch + ", Elapsed: " + string(D))
        drawnow
    end
end
```

Modelloss函数

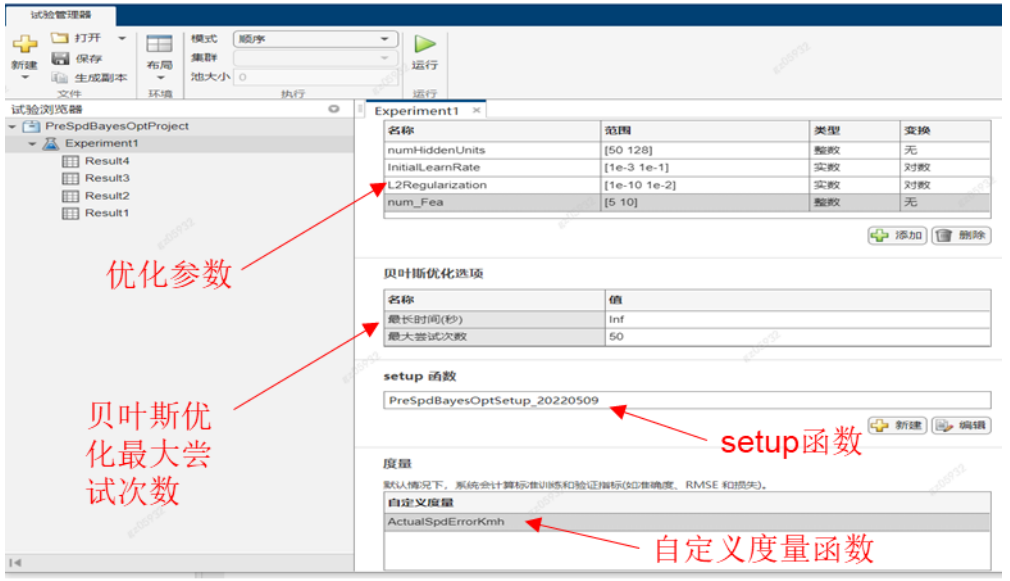
自定义训练循环

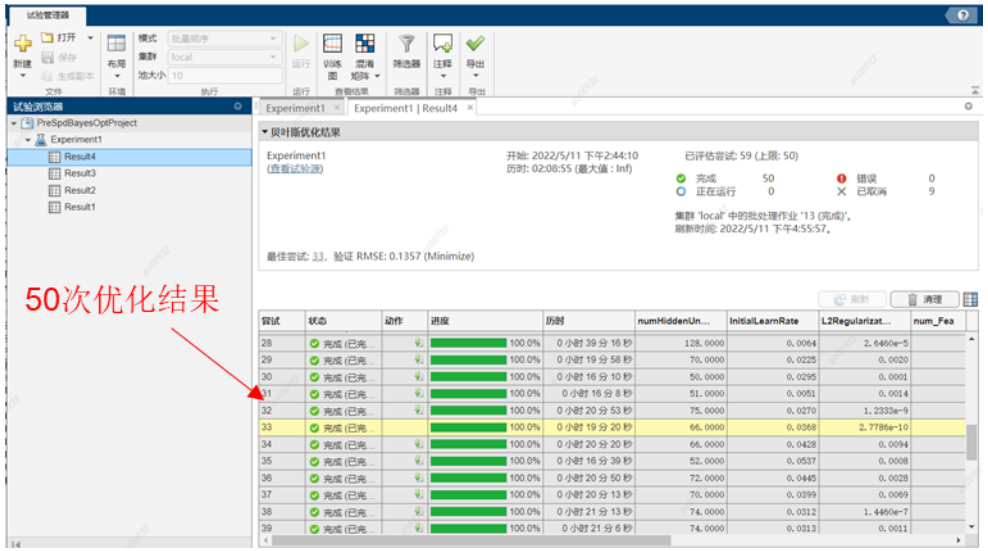


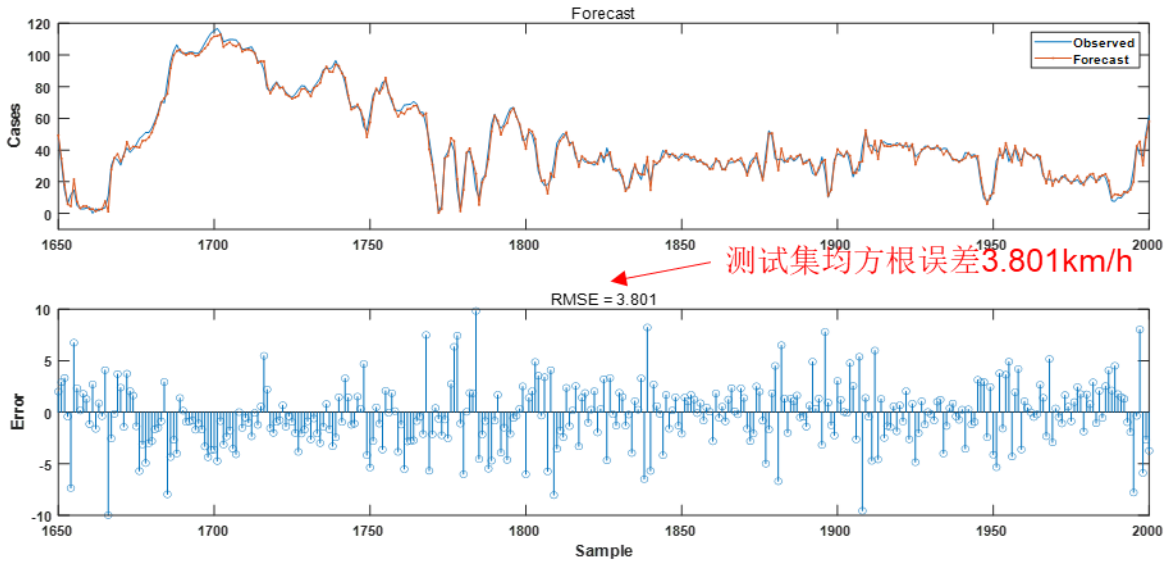
Seq2One速度预测模型

超参数优化

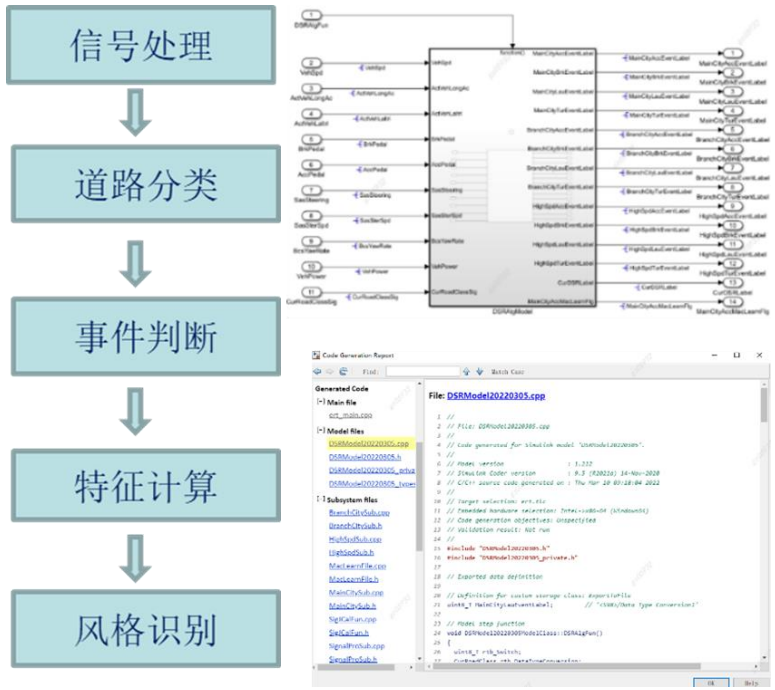
- 实验管理器app做超参数优化



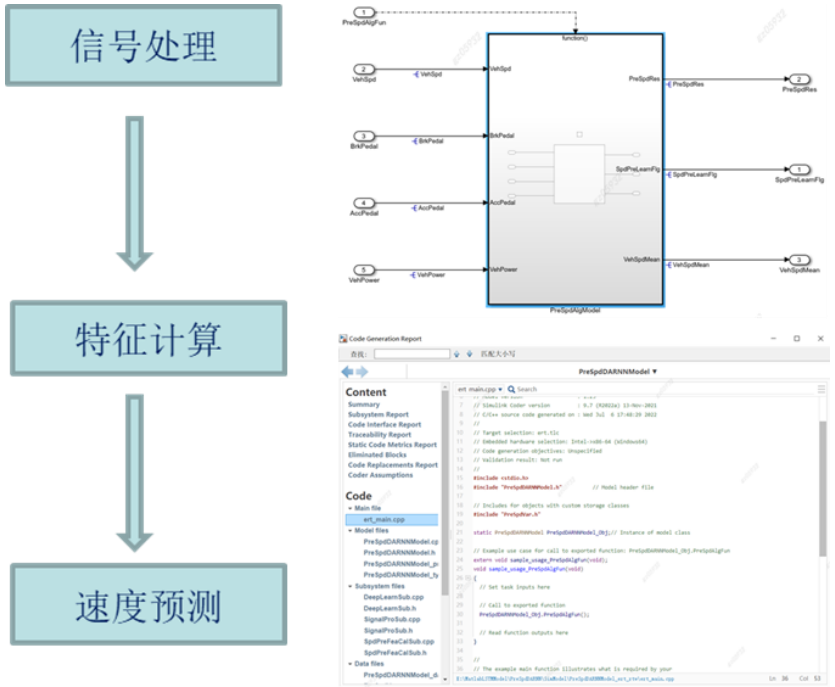




MBD开发



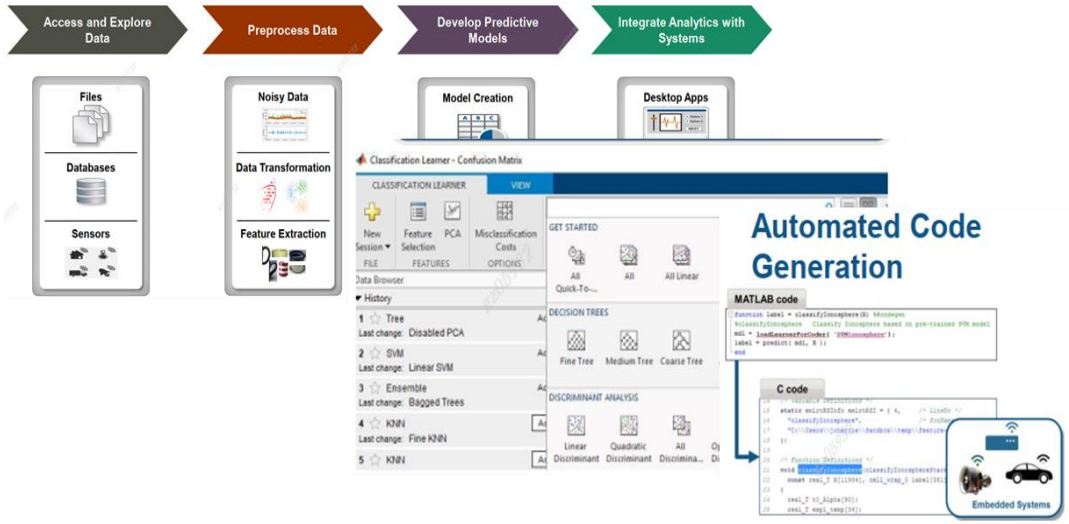
驾驶风格识别APP



速度预测APP

总结

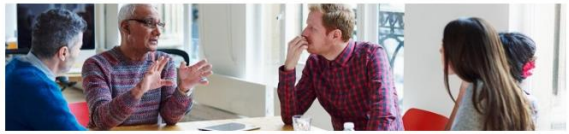
- 在MATLAB产品和团队支持下，我们团队快速实现了由传统嵌入式控制软件开发到数据科学应用的转型；
- 涉及方向：大数据处理、数据挖掘、机器学习和深度学习；




Training


Guided Evaluations


Onsite Workshops


Consulting


Technical Support

2022 MathWorks
中国汽车年会

Thank you