EVRI-thing You Need to Know About Diviner and PLS_Toolbox/Solo 9.5

Manny Palacios, Bob Roginski and Barry M. Wise

EIGENVECTOR RESEARCH INCORPORATED

Notes

- Webinar is being recorded and will be posted here:
  – https://eigenvector.com/resources/webinars/
- Scott Koch will act as moderator
- Please stay muted
- We don't generally use video
- Post questions in the chat
  – Questions will be relayed from moderator to speaker or
  – Answered directly in the chat

2

EIGENVECTOR RESEARCH INCORPORATED

Version 9.5

- Released in September 2024
- Main features:
  - Diviner, semi-automated machine learning tool for accelerating calibration model development
  - Ensemble models
- Release notes:

https://www.wiki.eigenvector.com/index.php?title=Release_Notes_Version_9_5

3

EIGENVECTOR RESEARCH INCORPORATED

The Problem

- **Too many choices** in model development
  - Meta parameters (#LVs etc.)
  - Preprocessing (Normalizations, derivatives, etc.)
  - Variable selection
- **Too many models** to build and update
- **Limited resources** of analyst's time and availability

EIGENVECTOR RESEARCH INCORPORATED

Our Goal

- **Speed up the development** of quality multivariate calibration models.
- **Transparency is key**. The analyst should be able to understand why the modeling choices were made.
- Use the user's **knowledge of data as an advantage**, compared to a black box approach.

We call it **Diviner**

EIGENVECTOR RESEARCH INCORPORATED

DIVINER – What is it?

Diviner is a semi-automated machine learning (Semi-AutoML) tool specifically designed to enhance the development of linear multivariate regression models. Unlike traditional AutoML systems that entirely automate the machine learning workflow — often sacrificing domain-specific insights and transparency — Diviner strikes a balance between automation and expert involvement. It allows users to efficiently leverage automation while retaining control over critical decision points in the modeling process. This hybrid approach addresses key shortcomings of AutoML, such as the lack of domain knowledge, overfitting, and limited customization, by incorporating user input to guide model development more effectively.

Diviner is currently configured for calibrating linear models, specifically Partial Least Squares (**PLS**) and regularized multiple linear regression (**MLR**) models, such as Elastic Net. It provides a comprehensive workflow that includes outlier assessment, a grid search for preprocessing methods, variable selection, and user-guided model refinement.

With its extensive library of preloaded preprocessing methods, Diviner is particularly suited for chemometrics and spectral data analysis. However, users can also create custom preprocessing libraries, making Diviner a versatile tool for linear regression tasks across various data types.

EIGENVECTOR RESEARCH INCORPORATED

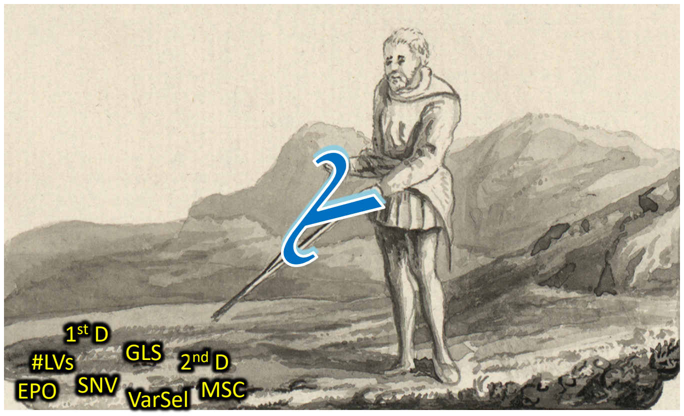
Diviner

Divine—to discover or locate something by intuition, insight or supernatural means.

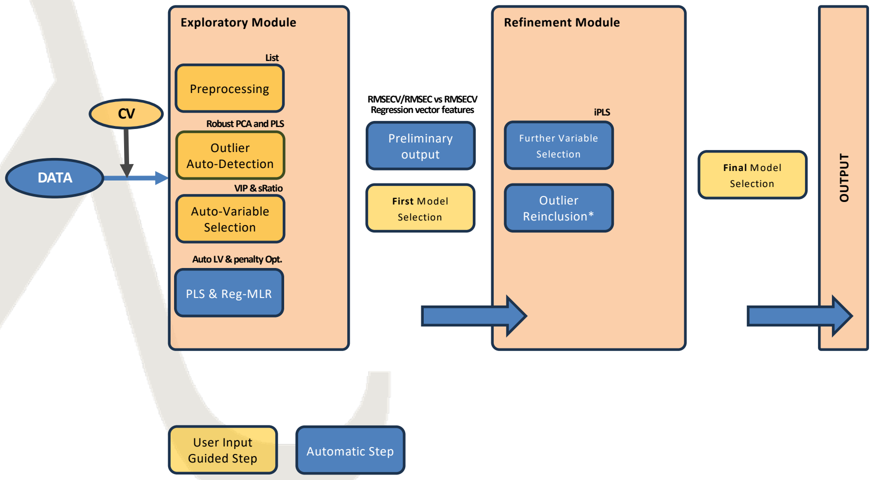
Not to replace the analyst, but rather to improve

Non pas pour remplacer l'analyste, mais plutôt pour améliorer

In French Deviner == to guess



The Diviner Workflow for Linear Regression



Diviner Roadmap

- Adding ensemble modeling
  - Currently only available only in the PLS_Toolbox
- Fully automated mode
- Numerous minor enhancements (from feedback)

- Non-linear regression modeling
- Classification (linear and non-linear)

EIGENVECTOR RESEARCH INCORPORATED

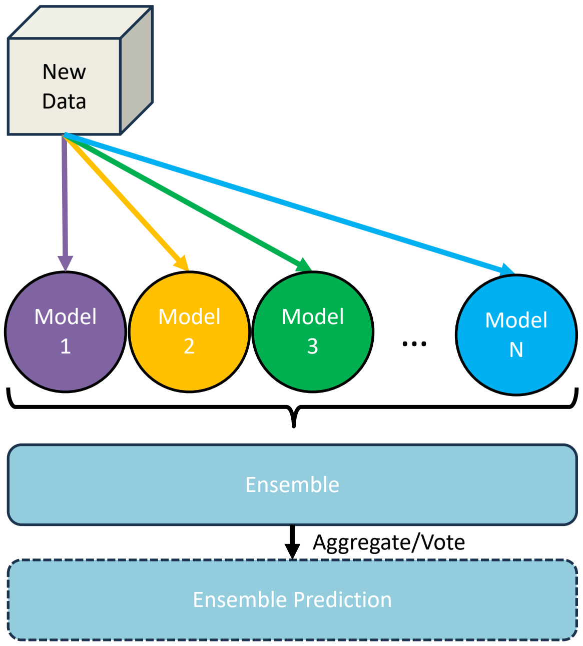
Key Takeaways

- **Redefining AutoML**: Limitations in AutoML's transparency and flexibility led to the creation of **Diviner, a Semi-AutoML solution**.
- **Balancing Automation and User Input**: Semi-AutoML approach balances full automation and user-led customization, harnessing both strengths.
- **Prioritizing Transparency and Customizability:** One-size-fits-all approach is insufficient. The future lies in workflows that can be tailored to unique project or domain needs.
- **User-Driven Model Selection:** Instead of one "best" model, explores a "family" of models. Increases likelihood of identifying the most accurate and robust (ideal) model for real-world deployment.
- **You might actually learn something from using it!**
- **Available in PLS_Toolbox/Solo 9.5**

EIGENVECTOR RESEARCH INCORPORATED

Ensembles

- Model-type that contains 2 or more **child** models.
  - Child models can be of any type
- Generalize better on new data using several high-performing models
- Ensemble predictions are an aggregation of child model predictions



Ensembles

Official definition: a group producing a single effect

- Democratic government systems
  - Electorates and laws are decided
- Ratings and reviews
  - Restaurants, hotels, products
- Music
  - Bands, orchestras, choir
- Clothing outfits
  - Shirt, pants, etc.

We all ensemble everyday!

Motivation

- Often, no single model is the answer
- Can be difficult to choose one model for deployment
- Bundle multiple high-performance models that look at the data in different ways
  - High/low concentrations
  - Different selected regions
- Generalize to new data more accurately by minimizing variance coming from individual models

Typical Ensemble Frameworks

- Ensemble Learning
  - Decision Trees, Random Forests, XGBoost
  - Stacking
    - Several Model Predictions -> Final Model -> Final Prediction (opposed to taking aggregation)
  - Bagging
    - Build several models on random subsets of calibration data, aggregate predictions
    - Many flavors – Drawing w/ (Bagging) w/o replacement of samples (Pasting) or variables (Random Subspaces) or both (Random Patches)
  - Adaboost
    - Fit several weaker models (slightly better than random guessing) on repeatedly modified versions of the data, similar to rPLS
- **Voting, Model Fusion**
  - Take pre-built models, aggregate their prediction
  - Classification "Soft Vote"
    - Aggregate the probabilities from each model
  - Classification "Hard Vote"
    - The prediction is the Mode of the Predicted Class Labels

Building Ensembles in PLS_Toolbox

- Fusion consists of two steps
  1. Building **diverse**, high-quality models
  2. Aggregating their predictions

Properties to vary

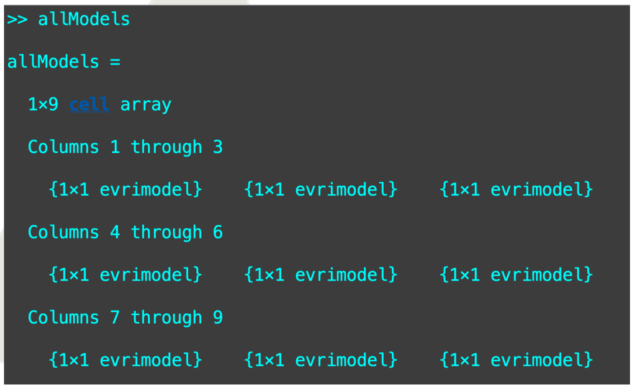
- Model-type
- Preprocessing
- Hyperparameters
- Included Variables
- Models that vary in performance at different concentration levels

Mean
Median
Jackknife

Diviner is a great way to build many quality, diverse models automatically!

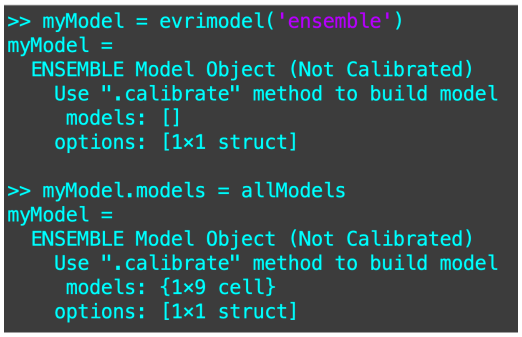
Building Ensembles in PLS_Toolbox

1. Building **diverse**, high-quality child models. Bundle into cell array



```
>> allModels

allModels =

  1×9 cell array

  Columns 1 through 3

    {1×1 evrimodel}    {1×1 evrimodel}    {1×1 evrimodel}

  Columns 4 through 6

    {1×1 evrimodel}    {1×1 evrimodel}    {1×1 evrimodel}

  Columns 7 through 9

    {1×1 evrimodel}    {1×1 evrimodel}    {1×1 evrimodel}
```

2. Construct and calibrate



```
>> myModel = evrimodel('ensemble')
myModel =
  ENSEMBLE Model Object (Not Calibrated)
    Use ".calibrate" method to build model
     models: []
    options: [1×1 struct]

>> myModel.models = allModels
myModel =
  ENSEMBLE Model Object (Not Calibrated)
    Use ".calibrate" method to build model
     models: {1×9 cell}
    options: [1×1 struct]
```

Model Fusion - Assumptions

1. All models are calibrated on same set of samples
2. RMSEC is an aggregate of the predicted values (fit) from each child
    1. .pred{2}
3. If all children are cross-validated, RMSECV is an aggregate of the cross-validated predicted values
    1. .detail.cvpred
4. RMSEP is an aggregate of children models applied to validation data

How to choose models for ensemble?

- Pick child models with best error
  - Okay to have some overfitting models…
- Keep diversity in mind
  - Preprocessing
  - Included Variables
  - Latent Variables
  - Model type
- How many children make the best ensemble?

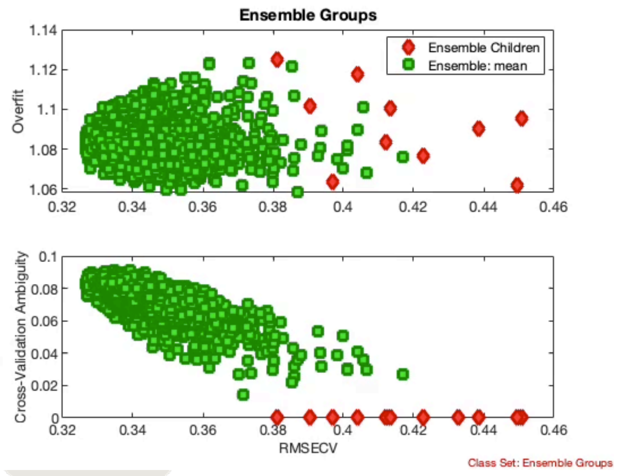
Answer: Let the computer find something!

Ensemblesearch

- Nchoosek-based algorithm to find ensembles with k models
- A minimum value and maximum value for k is provided
- Given a set of input models, combinations of these models will be built

```
INPUTS: [models,mink,maxk,aggregationfunc]
ALGORITHM:
combos = [];
for i=mink:maxk
 % create all possible combinations of ensemble from mink to maxk
 combos = [combos nchoosek(models,i)];
end
for i=1:size(combos,1)
 % obtain calibration, cross-validation predictions and ambiguities
end
% choose best ensemble based on RMSECV, Overfit, and ambiguity
```

Child Presence in Ensembles

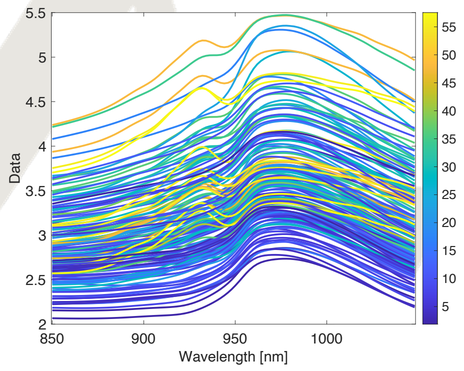


Example

The Tecator NIR data set contains spectra of 215 finely chopped pure meat samples using an Infratec Food and Feed Analyzer, providing the absorbance at 100 wavelengths across the
region 850-1050 nm. Also available are moisture, fat, and protein content in the meat.

Borggaard, Thodberg, Analytical Chemistry, 64 (1992) 545–551

http://lib.stat.cmu.edu/datasets/tecator



	RMSEC	RMSECV	RMSEP
PLS	1.77	1.91	1.87
LWR	0.41	0.72	0.77
SVM	0.41	**0.55**	0.58
ANN	0.45	0.61	0.63
ANNDL (sk)	0.23	0.68	**0.55**
ANNDL (tf)	0.63	0.75	0.85
XGB	0.02	1.76	1.70

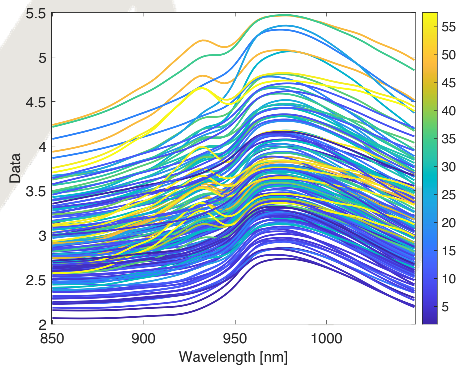
Can Ensemble do better?

Example

The Tecator NIR data set contains spectra of 215 finely chopped pure meat samples using an Infratec Food and Feed Analyzer, providing the absorbance at 100 wavelengths across the
region 850-1050 nm. Also available are moisture, fat, and protein content in the meat.

Borggaard, Thodberg, Analytical Chemistry, 64 (1992) 545–551

http://lib.stat.cmu.edu/datasets/tecator



	RMSEC	RMSECV	RMSEP
PLS	1.77	1.91	1.87
LWR	0.41	0.72	0.77
SVM	0.41	**0.55**	0.58
ANN	0.45	0.61	0.63
ANNDL (sk)	0.23	0.68	**0.55**
ANNDL (tf)	0.63	0.75	0.85
XGB	0.02	1.76	1.70
Ensemble	**0.40**	**0.52**	**0.53**

Can Ensemble do better?

Wrap Up

- Ensembles are a new model type introduced in version 9.5
- Currently supported algorithm is model fusion
- Consistently adds slight boost in performance
- Best performing ensembles are ones with lowest error and highest ambiguity
- High-performing, overfitting individual models can be balanced out by lower-performing, non-overfitting models in an ensemble
- Currently available from the MATLAB command line only
- Works with Solo_Predictor (coming soon)

How to Get Diviner and Ensembles

- Available in version 9.5 to users with current maintenance
  - Diviner now part of PLS_Toolbox and Solo
  - Ensembles not supported in Solo yet
- Out of maintenance? Renew now!