



Implementarea modelelor ca servicii web în ML Studio

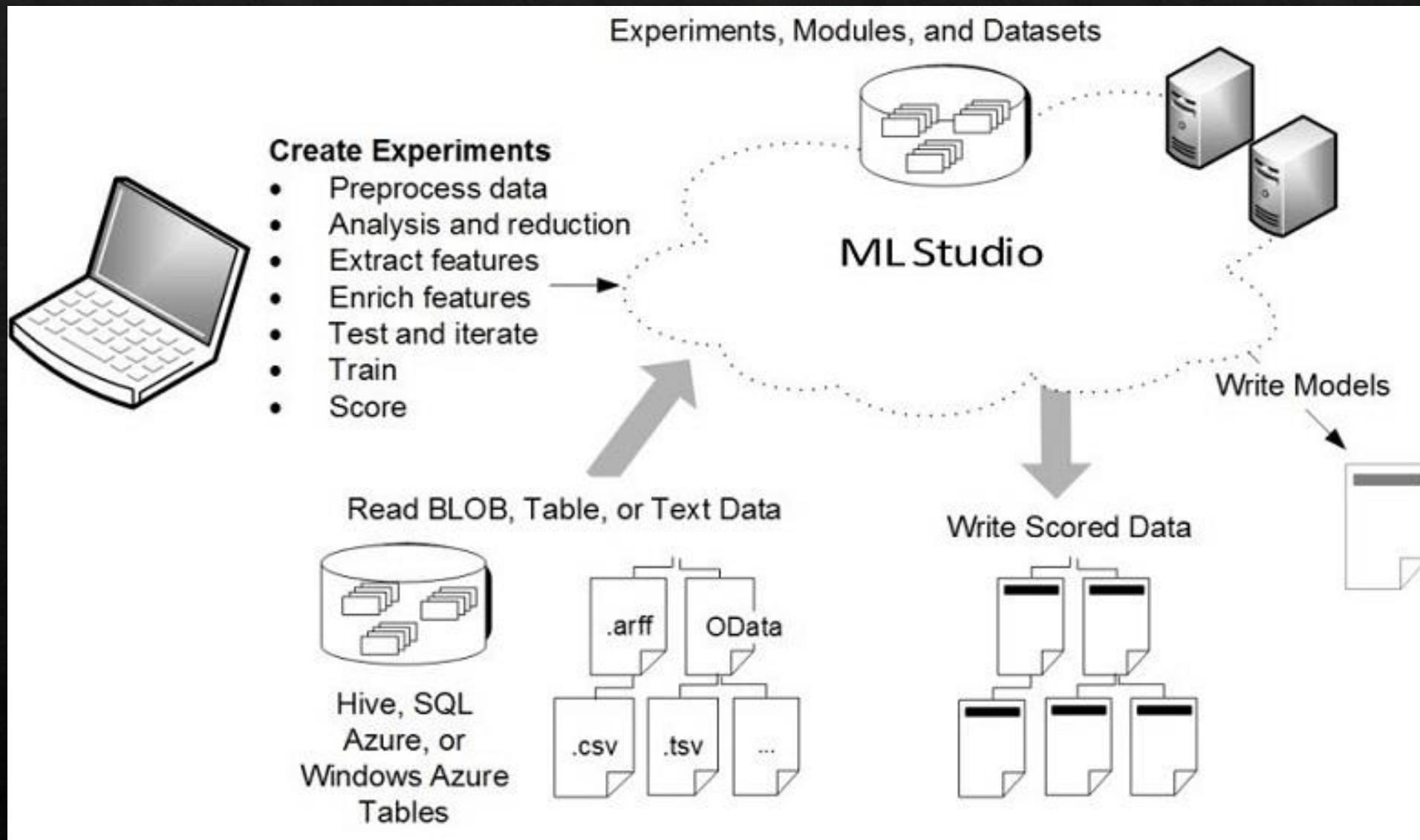
Cristian KEVORCHIAN

Facultatea de Matematică și Informatică

ML Studio(classic)

- ❖ Machine Learning Studio (clasic) este un instrument drag-and-drop care poate fi utilizat pentru a dezvolta, testa și implementa modele de învățare automată.
- ❖ ML Studio (clasic) facilitează publicarea modelelor ca servicii web, care pot fi consumate cu ușurință de aplicații personalizate sau de instrumente BI, cum ar fi Excel.

ML Studio un workspace interattivo



Machine Learning in ML Studio

Anomaly Detection

- One-class Support Vector Machine
- Principal Component Analysis-based Anomaly Detection
- Time Series Anomaly Detection*

Classification

Two-class Classification

- Averaged Perceptron
- Bayes Point Machine
- Boosted Decision Tree
- Decision Forest
- Decision Jungle
- Logistic Regression
- Neural Network
- Support Vector Machine

Multi-class Classification

- Decision Forest
- Decision Jungle
- Logistic Regression
- Neural Network
- One-vs-all

Clustering

- K-means Clustering

Recommendation

- Matchbox Recommender

Regression

- Bayesian Linear Regression
- Boosted Decision Tree
- Decision Forest
- Fast Forest Quantile Regression
- Linear Regression
- Neural Network Regression
- Ordinal Regression
- Poisson Regression

Statistical Functions

- Descriptive Statistics
- Hypothesis Testing T-Test
- Linear Correlation
- Probability Function Evaluation

Text Analytics

- Feature Hashing
- Named Entity Recognition
- Vowpal Wabbit

Computer Vision

- OpenCV Library

Data/Model Visualization

- Scatterplots
- Bar Charts
- Box plots
- Histogram
- R and Python Plotting Libraries
- REPL with Jupyter Notebook
- ROC, Precision/Recall, Lift
- Confusion Matrix
- Decision Tree*

Training

- Cross Validation
- Retraining
- Parameter Sweep

<https://studio.azureml.net>

Guest Access Workspace: Free trial access without logging in.

Free Workspace: Free persisted access, no Azure subscription needed.

Standard Workspace: Full access with SLA under an Azure subscription.

Cross browser drag & drop ML workflow designer.
Zero installation needed.

Import Data

Unlimited Extensibility

- R Script Module
- Python Script Module
- Custom Module
- Jupyter Notebook

Preprocess

Built-in ML Algorithms

Split Data

Train Model

Score Model

Training Experiment

One-click Operationalization

Predictive Experiment

Make Prediction with Elastic APIs

- Request-Response Service (RRS)
- Batch Execution Service (BES)
- Retraining API

Data Source

- Azure Blob Storage
- Azure SQL DB
- Azure SQL DW*
- Azure Table
- Desktop Direct Upload
- Hadoop Hive Query
- Manual Data Entry
- OData Feed
- On-prem SQL Server*
- Web URL (HTTP)

Data Format

- ARFF
- CSV
- SVMLight
- TSV
- Excel
- ZIP

Data Preparation

- Clean Missing Data
- Clip Outliers
- Edit Metadata
- Feature Selection
- Filter
- Learning with Counts
- Normalize Data
- Partition and Sample
- Principal Component Analysis
- Quantize Data
- SQLite Transformation
- Synthetic Minority Oversampling Technique

Enterprise Grade Cloud Service

- SLA: 99.95% Guaranteed Up-time
- Azure AD Authentication
- Compute at Large Scale
- Multi-geo Availability
- Regulatory Compliance*

Community

- Gallery (<http://gallery.azureml.net>)
- Samples & Templates
- Workspace Sharing and Collaboration
- Live Chat & MSDN Forum Support

Implementarea unui serviciu web de analiză predictivă

- ❖ Odată ce modelul este finalizat, acesta poate fi implementat ca un serviciu web de Machine Learning Studio (clasic).
- ❖ Serviciile web generate de ML Studio (clasic) oferă o interfață între o aplicație și un model de notare a fluxului de lucru Machine Learning Studio (clasic). O aplicație externă poate comunica cu fluxul de lucru al unui model MLStudio (clasic) în timp real.
- ❖ Un apel către serviciul web MLStudio (clasic) returnează rezultatele modelului către o aplicație externă.
- ❖ Pentru a efectua un apel către un serviciu web, este utilizată o cheie API care este creată odata cu implementarea serviciul web.
- ❖ Un serviciu web MLStudio (clasic) se bazează pe REST(REpresentational , State Transfer) o soluție arhitecturală pentru proiecte de programare web.

Tipuri de servicii

- ❖ Azure Machine Learning Studio (clasic) are două tipuri de servicii web:
 - ❖ Request-Response Service(RRS): un serviciu cu latență scăzută, foarte scalabil, care înregistrează o singură înregistrare de date.
 - ❖ Batch Execution Service (BES):un serviciu asincron care marchează un lot de înregistrări de date. Intrarea pentru BES este ca intrarea de date pe care RRS o folosește.

Principala diferență este că BES citește un bloc de înregistrări dintr-o varietate de surse, cum ar fi stocarea Blob Azure, stocarea Azure Table, baza de date SQL Azure, HDInsight (interrogare de tip stup) și surse HTTP.

Etapele implementării

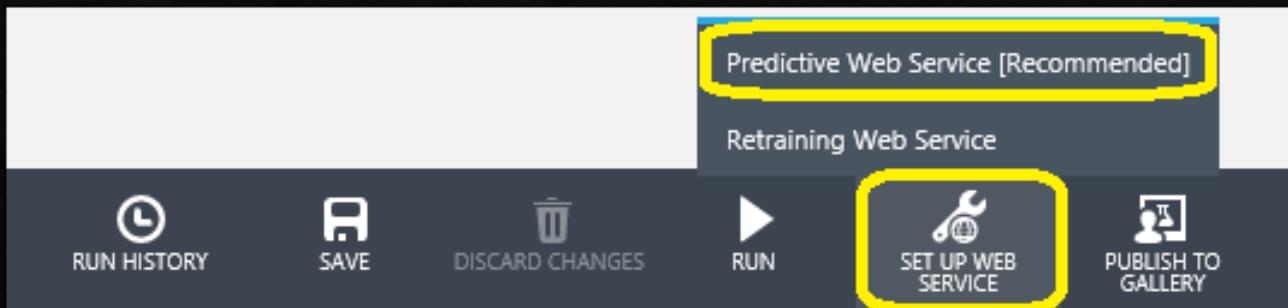
- ❖ Crearea un experiment de training - este posibilă antrenarea și testarea modelului folosind datele de training pe care le furnizam, utilizând un set mare de algoritmi de învățare automată.
- ❖ Transformarea într-un experiment operațional - Odată ce modelul a fost instruit cu datele existente și este gata să fie utilizat pentru lucrul cu date noi prin simplificarea experimentului.
- ❖ Se implementează ca un serviciu web nou sau ca un serviciu web clasic - Când implementați experimentul ca serviciu web Azure, utilizatorii pot trimite date modelului și pot primi output-ul acestuia.

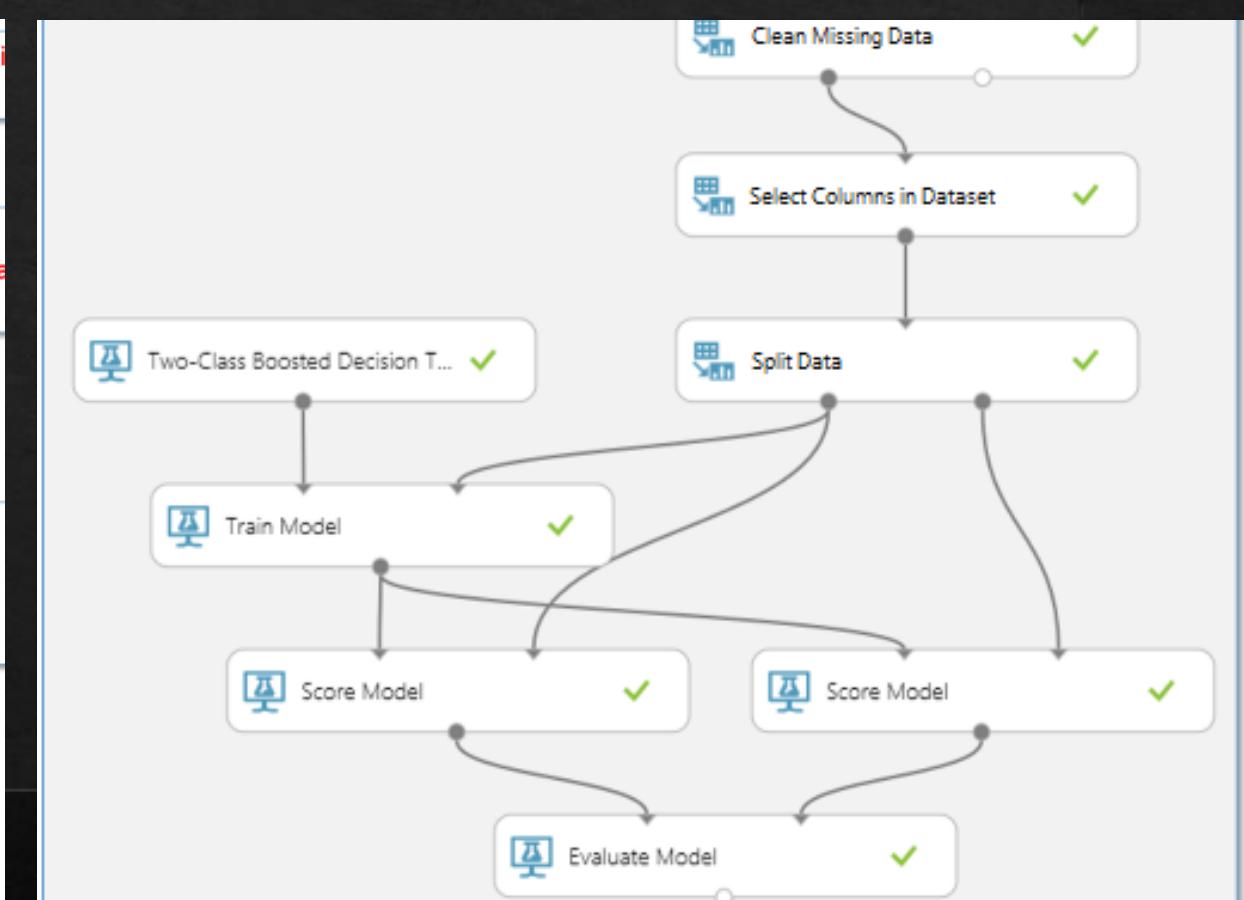
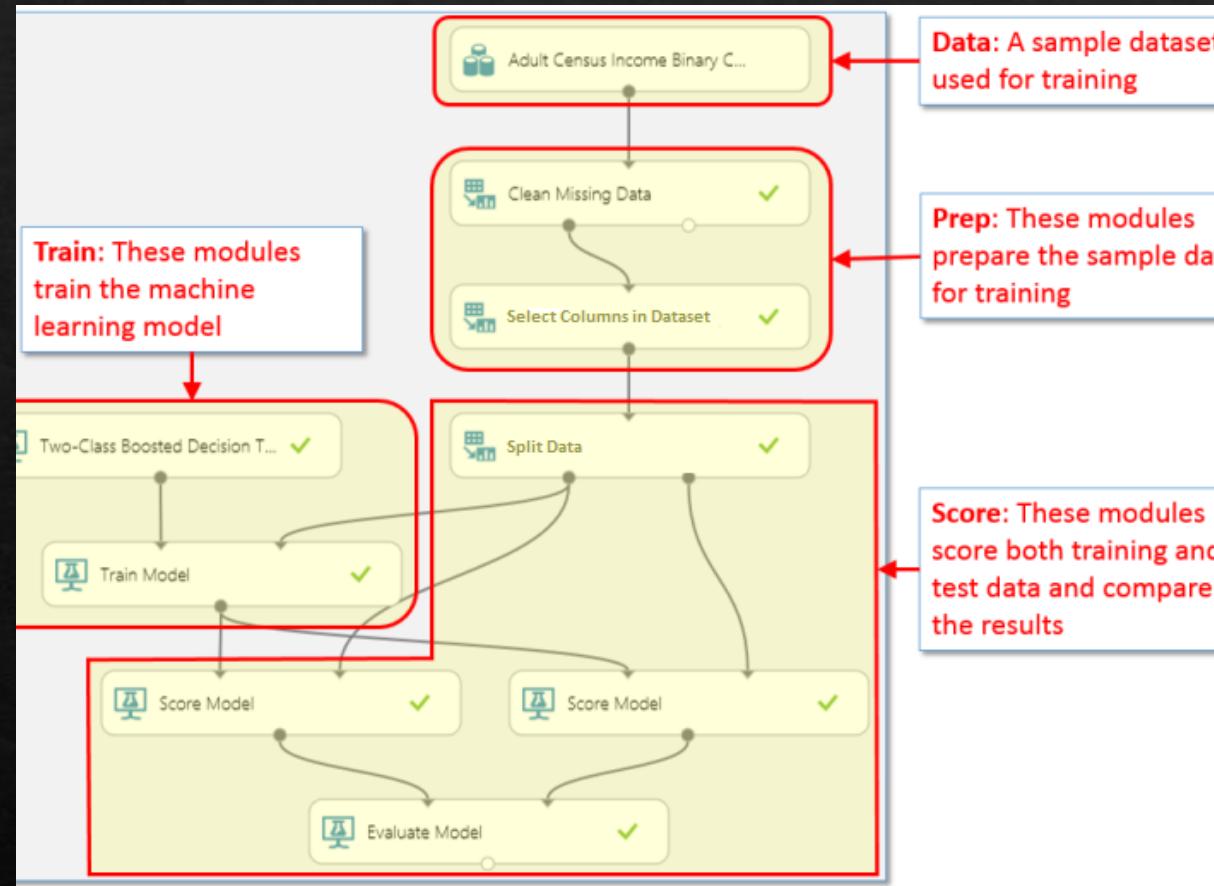
Convertirea experimentului de formare într-un experiment predictiv

- ❖ Odată cu instruirea modelului poate fi realizată convertirea experimentului de antrenament într-un experiment predictiv pentru a obține date noi. Trecând la un experiment predictiv, vă pregătiți modelul instruit pentru a fi implementat ca serviciu web.
- ❖ Utilizatorii serviciului web pot trimite date de intrare către model, iar modelul va furniza înapoi rezultatele predicției.
- ❖ Procesul de conversie a unui experiment de antrenare într-un experiment predictiv implică trei pași:
 - ❖ Înlocuirea modulelor algoritmului de învățare automată cu modelul instruit.
 - ❖ Reducerea experimentului doar la acele module necesare pentru scoring.
 - ❖ Definirea modul în care modelul va accepta datele de la utilizatorului serviciului web și ce date vor fi returnate.

REZUMAT

- ❖ În experimentul de formare, antrenam și generăm scoring asociat modelului folosind propriile date. Odată implementate, utilizatorii vor încărca noi date în model și acesta va furniza rezultatele predicției.
- ❖ Prin urmare, pe măsură ce este convertit experimentul de antrenament într-un experiment predictiv pentru al pregăti pentru implementare, se definește modul în care modelul va fi folosit de utilizatori.





Implementarea exclude MODULE

- ❖ Când convertim un experiment de formare într-un experiment predictiv, unele dintre module nu mai sunt necesare :
- ❖ Date - Trebuie păstrat setul de date din experimentul predictiv, astfel încât acesta să poată furniza metadatele necesare noilor date adăugate.
- ❖ Pregătirea datelor - În funcție de datele utilizatorului care vor fi utilizate pentru scoring, aceste module pot fi sau nu necesare pentru procesarea datelor primite.
- ❖ Antrenare - La configurarea serviciului web, aceste module sunt înlocuite cu un singur modul care conține modelul instruit. Acest nou modul este salvat în secțiunea Modele antrenate din secțiun de module.
- ❖ Scor - În acest exemplu, modulul Split Data este utilizat pentru a împărți fluxul de date în date de testare și date de antrenament. În experimentul predictiv, nu mai antrenăm, astfel încât Split Data poate fi eliminat. În mod similar, al doilea modul de scoring și modulul de evaluare model sunt utilizate pentru a compara rezultatele, astfel încât aceste module nu sunt necesare în experimentul predictiv. Cu toate acestea, modulul de scoring al modelului rămas este necesar pentru a returna un rezultat de scor prin intermediul serviciului web.

