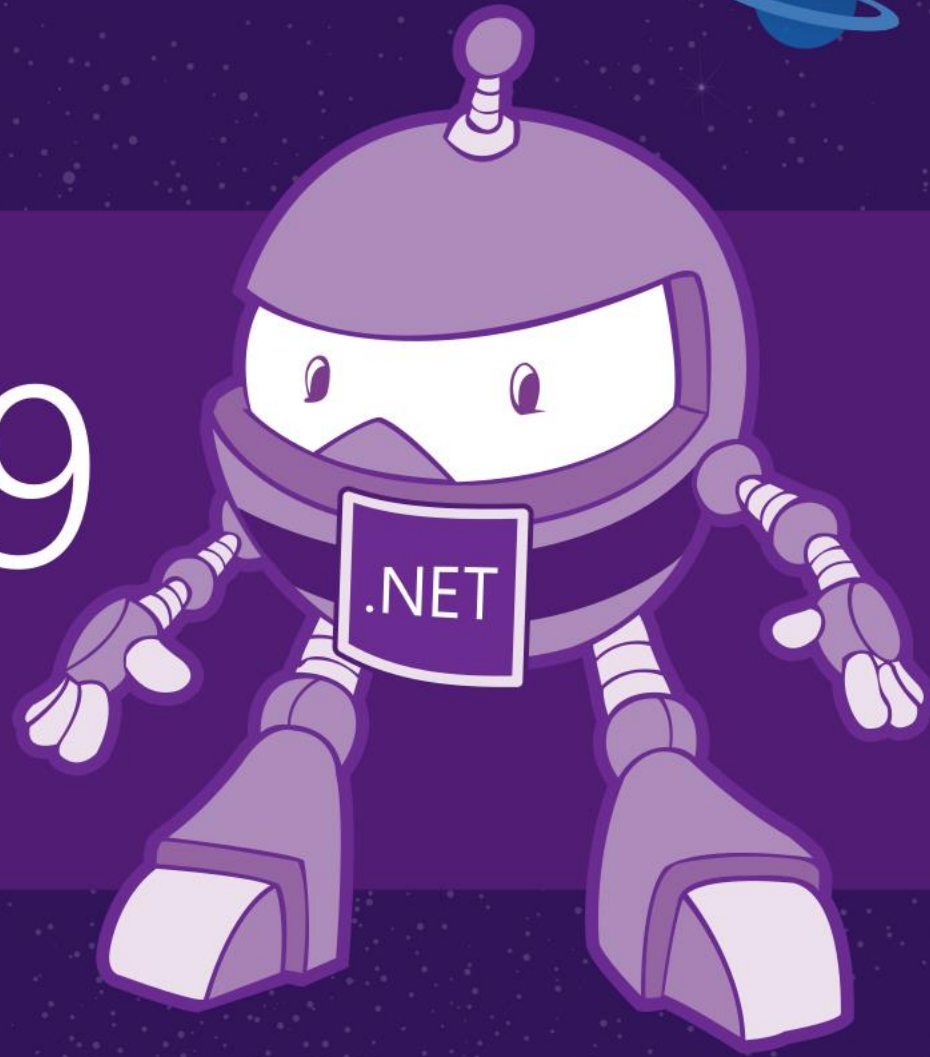


# .NET Conf 2019

Discover the world of .NET



[www.dotnetconf.net](http://www.dotnetconf.net)

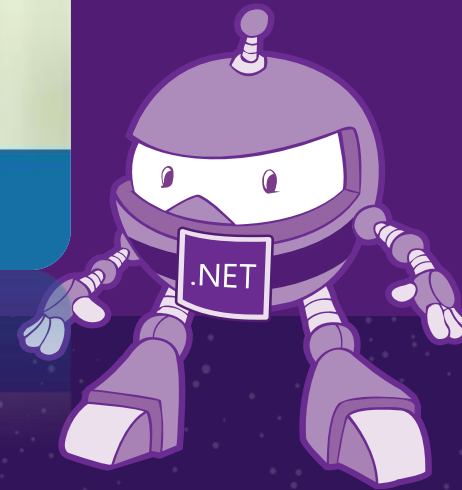
# What's new in ML.NET?

(Machine Learning for .NET)

Miguel Arturo Valle Pelaez

Microsoft MVP IA

@mavpelaez



# About: Miguel Aturo Valle Pelaez

Microsoft MVP IA

Director EEQIS SAC

Director SQL PASS Chimbote

Director de Capitulo I-S-C CAD Chimbote

Director General Instituto BITEC

Maestría Gerencia de Operaciones y Logística UPC

Maestrando MBA Centrum Catolica y Master Internacional EADA

Maestría UPAO Sistemas de Información

Master Liderazgo Gerencial

Machine learning, Business Intelligence Oberta Cataluña

Diplomado en Gerencia de Proyectos UPC

Diplomado en Investigación Científica

Diplomado en Soluciones Informáticas

Ingeniero Informático y de sistemas

Catedrático universitario a tiempo completo con 12 años de experiencia dictando

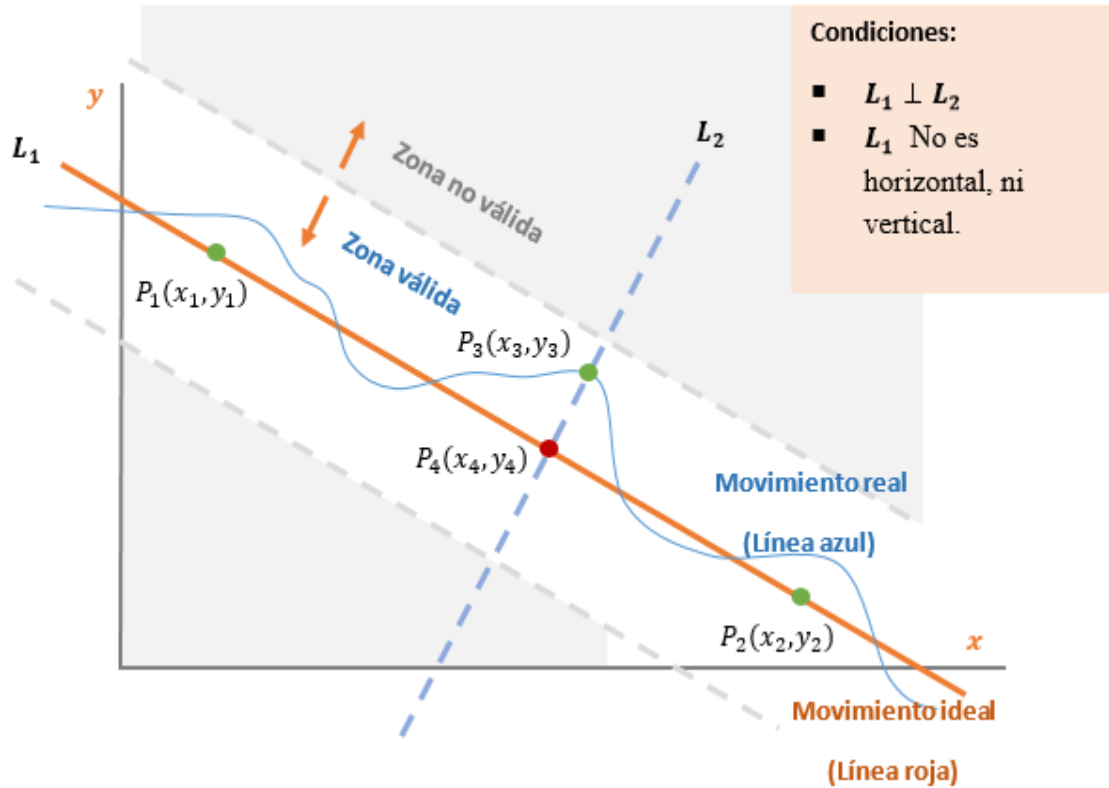
Microsoft 16 años de experiencia en la Arquitectura de TI



# Experiencia previa lenguaje y ML



# Matemática física en ML



SOLUCIÓN:

Pendiente de  $L_1$ :

$$m_1 = \frac{y_2 - y_1}{x_2 - x_1}, \quad m_1 \in L_1$$

Recta perpendicular a  $L_1$

$$L_1 \perp L_2$$

$$m_2 = -1 \cdot (m_1)^{-1}$$

$$m_2 = \frac{x_1 - x_2}{y_2 - y_1}, \quad m_2 \in L_2$$

Ecuación de las rectas  $L_1$  y  $L_2$

$$L_1: y - y_1 = m_1(x - x_1) \quad \dots (1)$$

$$L_2: y - y_3 = m_2(x - x_3) \quad \dots (2)$$



# Lenguaje ML

```
// Efecto magnético:
// Algoritmo aplicando ecuacion de la recta para que el puntero se mueva conforme la linea:
double x1, x2, x;
double y1, y2, y;
double delta;

x1 = _linea[index].X1;
x2 = _linea[index].X2;
y1 = _linea[index].Y1;
y2 = _linea[index].Y2;
double m1, m2;
if (Math.Abs(x1 - x2) > 0 && Math.Abs(y1 - y2) > 0) // Diagonal
{
    m1 = (y2 - y1) / (x2 - x1); // Pendiente normal
    m2 = (x1 - x2) / (y2 - y1); // Pendiente perpendicular
    x = (m1 * x1 - m2 * px - y1 + py) / (m1 - m2);
    x = (x - 8) * 1.1; // Ajuste maximizar 'x'

    y = m1 * (x - x1) + y1;
    delta = ((py - y) / 5);
    if (delta < -30 || delta > 15) { this.SoltarImagen(); return; }
    y = y + delta; // Ajuste delta en 'y'

    Canvas.SetLeft(mypointer, x - mypointer.ActualWidth / 2);
    Canvas.SetTop(mypointer, y - mypointer.ActualHeight / 2);

    this.SoltarImagen(x, y);
    if (_scopeCount > 2) this.NuevaRonda();
}
else if (Math.Abs(y1 - y2) <= 0) // Horizontal
{
    x = (px - 8) * 1.1; // Ajuste maximizar 'x'
    delta = ((py - y1) / 5);
    if (delta < -30 || delta > 15) { this.SoltarImagen(); return; }
    y = y1 + delta; // Ajuste delta en 'y'

    Canvas.SetLeft(mypointer, x - mypointer.ActualWidth / 2);
    Canvas.SetTop(mypointer, y - mypointer.ActualHeight / 2);

    this.SoltarImagen(x, y);
    if (_scopeCount > 2) this.NuevaRonda();
}
else if (Math.Abs(x1 - x2) <= 0) // Vertical
{
    x = x1 + ((px - x1) / 5);
    y = (py - 8) * 1.1;
    Canvas.SetLeft(mypointer, x - mypointer.ActualWidth / 2);
    Canvas.SetTop(mypointer, y - mypointer.ActualHeight / 2);
}
```

# Proceso efecto magnetico



# Lenguaje F#

The screenshot shows the Microsoft Visual Studio IDE with the following components:

- Title Bar:** DescribePicture - Microsoft Visual Studio
- Menu Bar:** File, Edit, View, Project, Build, Debug, Team, Tools, Architecture, Test, .NET Reflector, R Tools, Analyze, Window, Help
- Toolbar:** Includes icons for opening files, saving, undo, redo, and a 'Start' button. It also shows 'Debug' and 'Any CPU' dropdowns.
- Left Sidebar:** Contains 'Server Explorer' and 'Toolbox'.
- Right Sidebar:** Contains 'Properties', 'Solution Explorer', 'Team Explorer', 'Class View', 'Notifications', '.NET Reflector', and 'Object Browser'.
- Editor Window:** Displays the file 'Learning.fsx'. The code is as follows:

```
let res3 =  
    Http.RequestString("https://datamarket.accesscontrol.windows.net/v2/OAuth2-13",  
        body = HttpRequestBody.FormValues [ "client_id", "democip-2016"  
        "client_secret", "WnoyJN3lx/xeDCmn5cEVbqD3zzL03iB3RGrTu9plwr0="  
        "scope", "http://api.microsofttranslator.com"  
        "grant_type", "client_credentials" ])  
  
type AuthToken = JsonProvider< @"C:\Users\MIRANDA\Desktop\democs\DescribePicture\translateTokenResult.json" >  
  
let jsonToken = AuthToken.Parse res3  
  
let translatedXml =  
    Http.RequestString("http://api.microsofttranslator.com/v2/Http.svc/Translate",  
        query = [ "text", pictureDescription  
        "from", "en"  
        "to", "es" ],  
        headers = [ "Authorization", "Bearer " + jsonToken.AccessToken ])  
  
type TranslationXml = XmlProvider< ""<string xmlns="http://schemas.microsoft.com/2003/10/Serialization/">¡Hola mundo!</string>"" >  
  
let translatedText = TranslationXml.Parse translatedXml  
//-----
```
- Bottom Bar:** Includes 'Error List', 'Output', 'Package Manager Console', and '.NET Reflector Analyzer'. The zoom level is set to 128%.



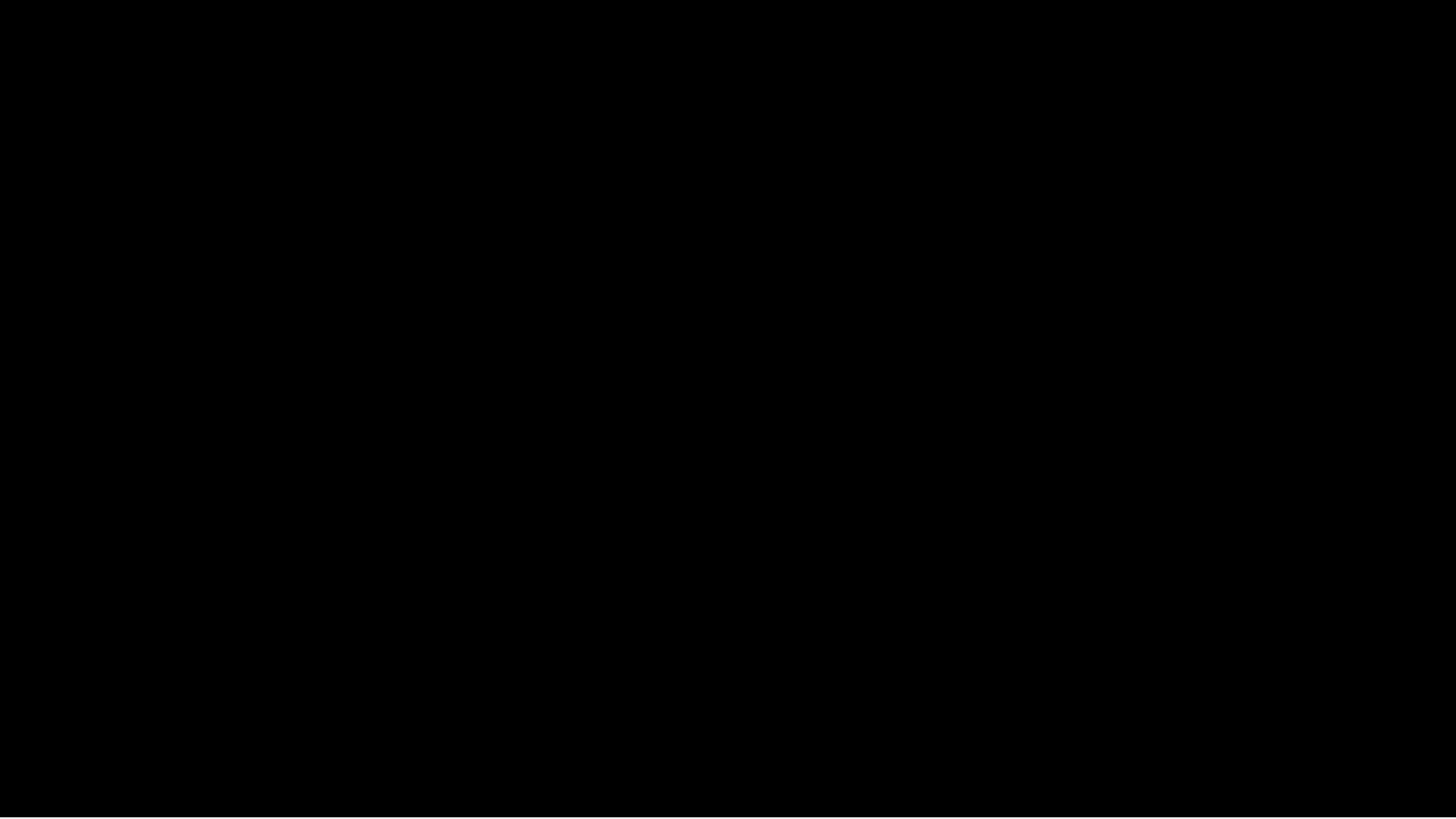


Como resultado nos dice que en la foto se observa una persona hombre sonriendo y una pequeña niña con una pelota

# Introduccion rapida en **ML.NET**

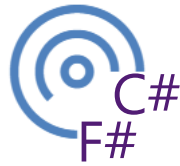
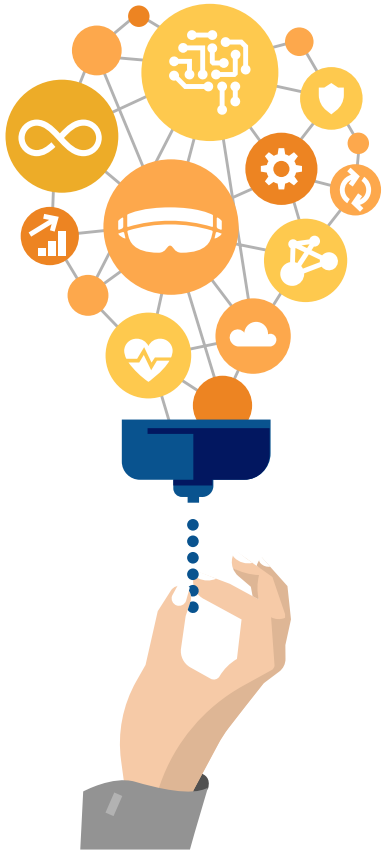
# .NET





# ML.NET

Un marco de aprendizaje automático de código abierto y multiplataforma



**Creado para  
desarrolladores  
de .NET**



**ML personalizado  
hecho fácil con  
herramientas**

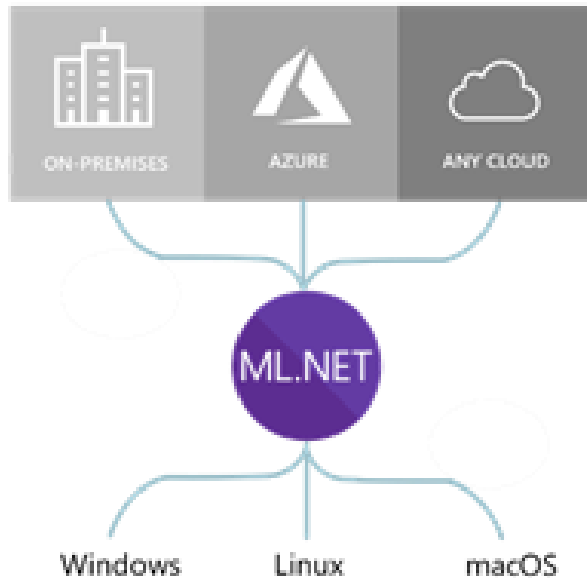


**Extendido con  
TensorFlow y más**



**De confianza y  
probado a escala**

# ML.NET corre en cualquier lugar



## Sorporta Frameworks:

.NET Core (*Natively*)

.NET Framework (*Natively*)

Python with *NimbusML* (*Python bindings*)

## Sorporta arquitectura procesador.

x64

x86



# Algunas cosas que puedes hacer con ML.NET ...



## Sentiment analysis

Analyze the sentiment of customer reviews using a binary classification algorithm.

[Sentiment analysis sample >](#)



## Product recommendation

Recommend products based on purchase history using a matrix factorization algorithm.

[Product recommendation sample >](#)



## Price prediction

Predict taxi fares based on distance traveled etc. using a regression algorithm.

[Price prediction sample >](#)



## Customer segmentation

Identify groups of customers with similar profiles using a clustering algorithm.

[Customer segmentation sample >](#)



## GitHub labeler

Suggest the GitHub label for new issues using a multi-class classification algorithm.

[GitHub labeler sample >](#)



## Fraud detection

Detect fraudulent credit card transactions using a binary classification algorithm.

[Fraud detection sample >](#)



## Spam detection

Flag text messages as spam using a binary classification algorithm.

[Spam detection sample >](#)



## Image classification

Classify images (e.g. broccoli vs pizza) using a TensorFlow deep learning algorithm.

[Image classification sample >](#)



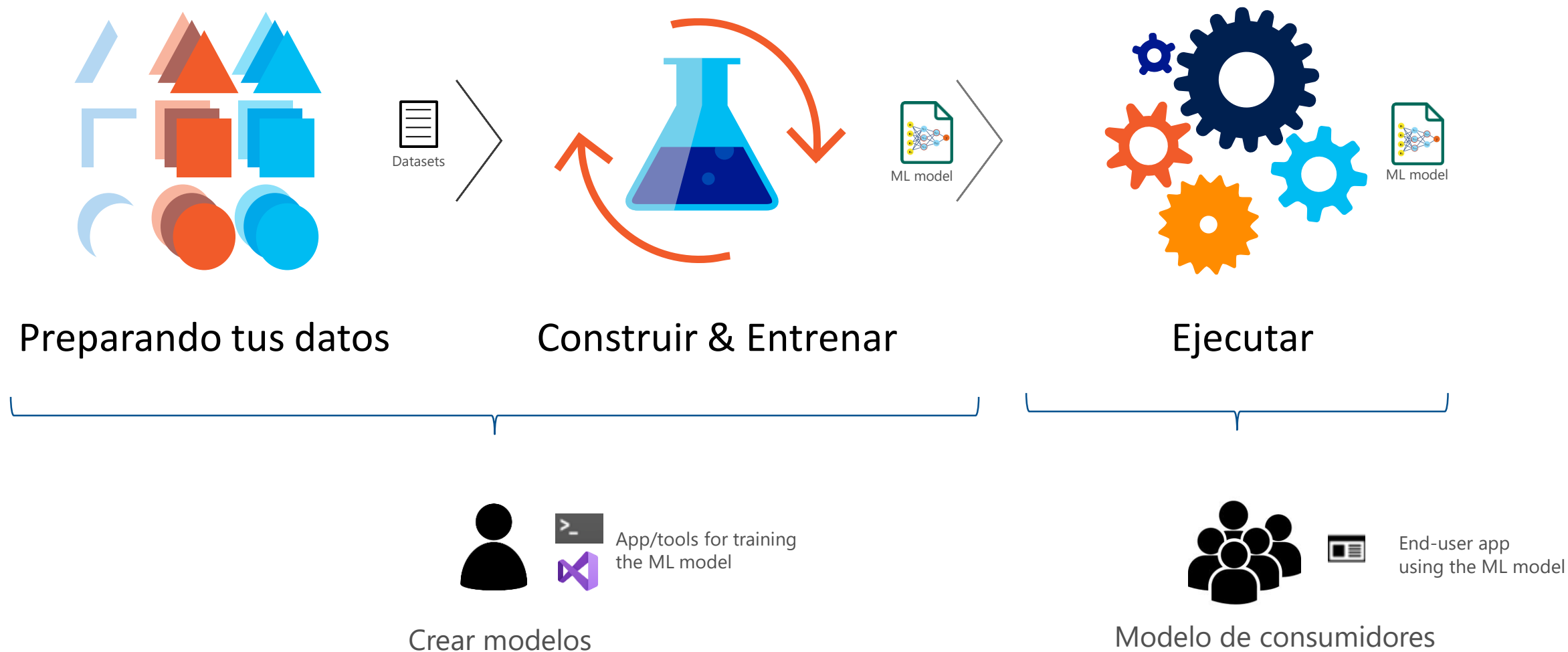
## Sales forecasting

Forecast future sales for products using a regression algorithm.

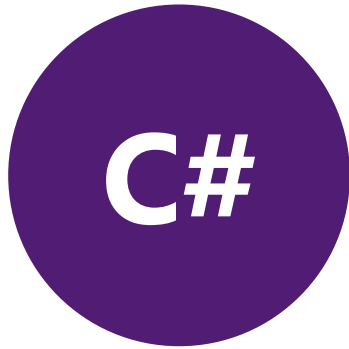
[Sales forecasting sample >](#)

<https://github.com/dotnet/machinelearning-samples>

# Flujo de trabajo Machine Learning



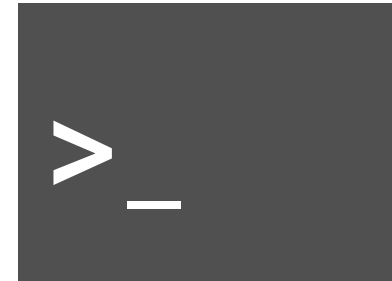
# Tres formas de usar ML.NET...



ML.NET  
**API**  
(Code)



ML.NET  
**Model Builder**  
(Visual Studio UI)



ML.NET  
**CLI**  
(Command-Line  
Interface)

# ML.NET esta probado para la batalla y listo para la empresa

## Productos Microsoft

(Usando ML.NET internamente hace > 8 años)



Bing - Ad Predictions



Excel - Chart Recommendations



Power Point - Design Ideas



Microsoft Defender – Antivirus Threat Protection



Azure ML Studio – Multiple components  
Azure Stream Analytics - Anomaly Detection

## Clientes

(ML.NET v1 desde mayo 2019)



Evolution Software – Hazelnut drying time prediction



Williams Mullen – **Law document classification**



Sig Parser – **E-mail spam detection**



Brenmor – Medical patient survey classification



endjin – Newsletter article classification

+ more

# La forma más fácil de crear un modelo ML.NET: **Construir modelos en Visual Studio**





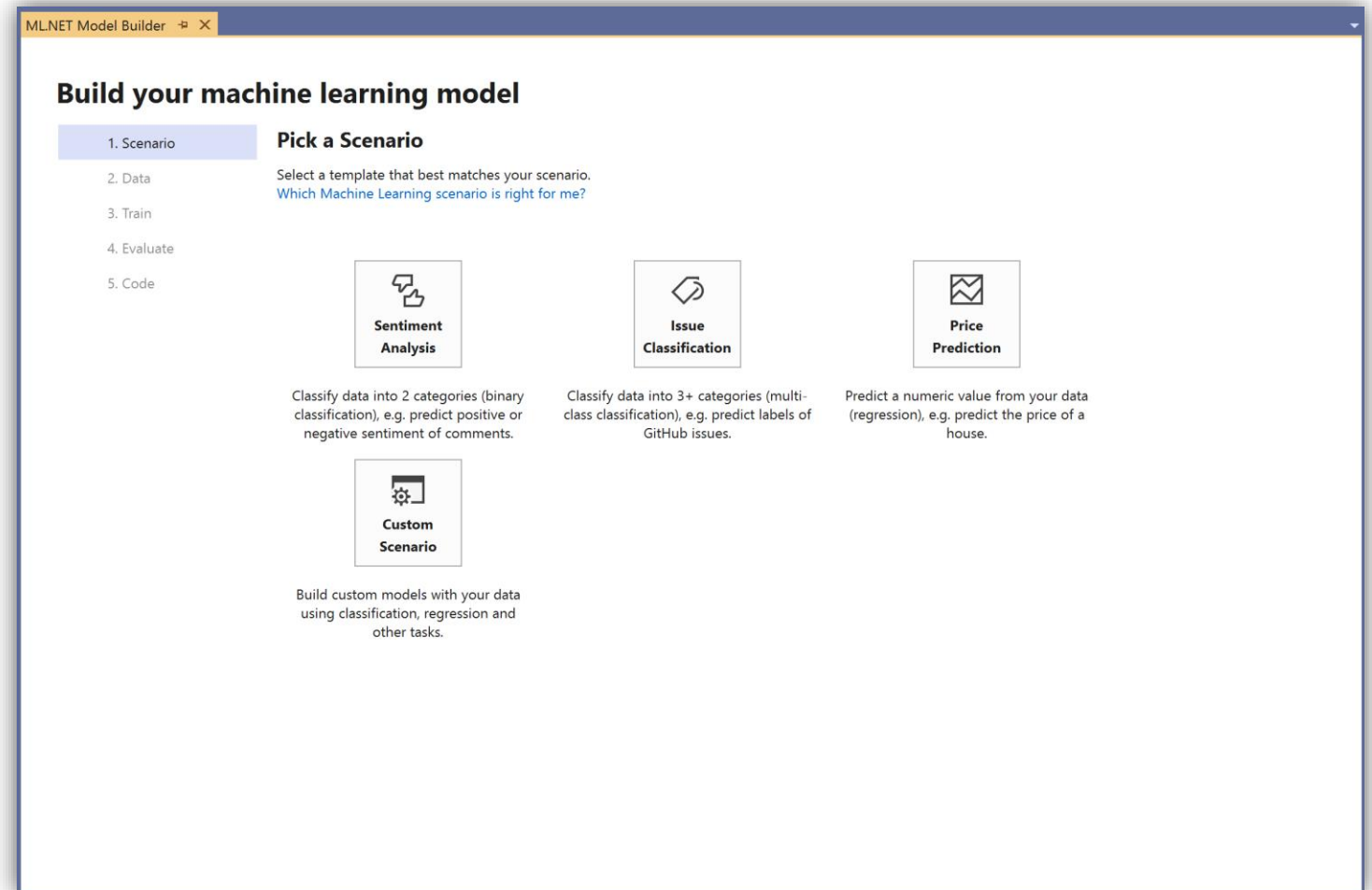
# ML.NET Construir Modelo

## Accesible machine learning in Visual Studio

- Una interfaz de usuario simple para construir fácilmente modelos ML personalizados con ML automatizado
- Cargar desde archivos y bases de datos
- Generar código para capacitación y consumo.
- Ejecuta todo local

Download VS vsix:

<http://aka.ms/mlnetmodelbuilder>



# Demo 1: **Model Builder en** Visual Studio .NET



File

Edit

View

Project

Build

Debug

Architecture

Test

Analyze

Tools

Extensions

Window

Help

Search (Ctrl+Q)

cnvml2

Live Share

ModelOutput.cs\*

ModelInput.cs\*

ML.NET Model Builder

1. Scenario

2. Data

3. Train

4. Evaluate

5. Code

Evaluate

Results of training for your model can be found below.

How do I understand my model performance?

Output

Overview

Details

ML Task:binary-classification

Training Time:4.6 seconds

Models Explored (Total):1 | [View Top 1 model explored](#)

Overall accuracy:

68.29%

Try your model

Note: Fields below are pre-filled by row 1 of your data.

SentimentText

DE== Dude, you are rude upload that carl picture back, or else.

Predict

Next Step: [Code](#)

Solution Explorer

Search Solution Explorer (Ctrl+)

Solution 'cnvml2' (3 of 3 projects)

cnvml2

Dependencies

cnvml2ML.ConsoleApp

cnvml2ML.Model

Program.cs

wikipedia-detox-250-line-data.tsv

cnvml2ML.ConsoleApp

Dependencies

ModelBuilder.cs

Program.cs

cnvml2ML.Model

Dependencies

ConsumeModel.cs

MLModel.zip

ModelInput.cs

ModelOutput.cs

Ready

Type here to search

6

# ML.NET CLI

- Construir fácilmente modelos ML personalizados con ML automatizado
- Multiplataforma (Windows, Linux, MacOS)
- Generar código para capacitación y consumo.
- Accesible

## macOS / Linux (Bash)

```
cesard@cli-test$ mlnet auto-train --task binary-classification --dataset "yelp_labelled.txt" --label-column-index 1 --has-header false --max-exploration-time 10
Exploring multiple ML algorithms and settings to find you the best model for ML task: binary-classification
For further learning check: https://aka.ms/mlnet-cli

Best Accuracy: 87.36%, Best Algorithm: LbfgsLogisticRegressionBinary, Last Algorithm: SgdCalibratedBinary
00:00:18

=====Experiment Results=====
Summary
-----
ML Task: binary-classification
Dataset: yelp_labelled.txt
Label: Label
Total experiment time : 18.60 Secs
Total number of models explored: 48

Top 5 models explored
-----
| Trainer | Accuracy | AUC | AUPRC | F1-score | Duration | #Iteration | |
|---|---|---|---|---|---|---|---|
| 1 | LbfgsLogisticRegressionBinary | 0.8736 | 0.9386 | 0.9272 | 0.8866 | 0.1 | 12 |
| 2 | SgdCalibratedBinary | 0.8736 | 0.9386 | 0.9408 | 0.8890 | 0.1 | 34 |
| 3 | AveragedPerceptronBinary | 0.8621 | 0.9255 | 0.9181 | 0.8667 | 0.9 | 1 |
| 4 | LbfgsLogisticRegressionBinary | 0.8621 | 0.9386 | 0.9381 | 0.8788 | 0.2 | 18 |
| 5 | LbfgsLogisticRegressionBinary | 0.8621 | 0.9313 | 0.9308 | 0.8798 | 0.1 | 24 |

Generated trained model for consumption: /Users/cesard/cli-test/SampleBinaryClassification/SampleBinaryClassification.Model\MLModel.zip
Generated C# code for model consumption: /Users/cesard/cli-test/SampleBinaryClassification/SampleBinaryClassification.ConsoleApp
Check out log file for more information: /Users/cesard/cli-test/SampleBinaryClassification/logs/debug_log.txt
cesard@cli-test$
```

## Windows (PowerShell and CMD)

```
Windows PowerShell
Exploring multiple ML algorithms and settings to find you the best model for ML task: binary-classification
For further learning check: https://aka.ms/mlnet-cli

Best Accuracy: 73.56%, Best Algorithm: SdcaLogisticRegressionBinary, Last Algorithm: LightGbmBinary
00:00:10

=====Experiment Results=====
Summary
-----
ML Task: binary-classification
Dataset: yelp_labelled.txt
Label: Label
Total experiment time : 10.60 Secs
Total number of models explored: 14

Top 5 models explored
-----
| Trainer | Accuracy | AUC | AUPRC | F1-score | Duration | #Iteration | |
|---|---|---|---|---|---|---|---|
| 1 | SdcaLogisticRegressionBinary | 0.7356 | 0.8122 | 0.8555 | 0.7527 | 0.3 | 13 |
| 2 | SdcaLogisticRegressionBinary | 0.7241 | 0.8048 | 0.8495 | 0.7447 | 0.5 | 2 |
| 3 | LightGbmBinary | 0.7126 | 0.7346 | 0.7818 | 0.7253 | 0.9 | 3 |
| 4 | LinearSvmBinary | 0.7126 | 0.7777 | 0.8088 | 0.7423 | 0.3 | 5 |
| 5 | FastTreeBinary | 0.7011 | 0.7569 | 0.7707 | 0.7174 | 2.1 | 6 |

Generated trained model for consumption: C:\cli-test\SampleBinaryClassification\SampleBinaryClassification.Model\MLModel.zip
Generated C# code for model consumption: C:\cli-test\SampleBinaryClassification\SampleBinaryClassification.ConsoleApp
Check out log file for more information: C:\cli-test\SampleBinaryClassification\logs\debug_log.txt
PS C:\cli-test>
```

```
> mlnet auto-train --ml-task binary-classification --dataset "customer-reviews.tsv" --label-column-name Sentiment
```

# Qué hay de nuevo para la lectura de datos

# .NET

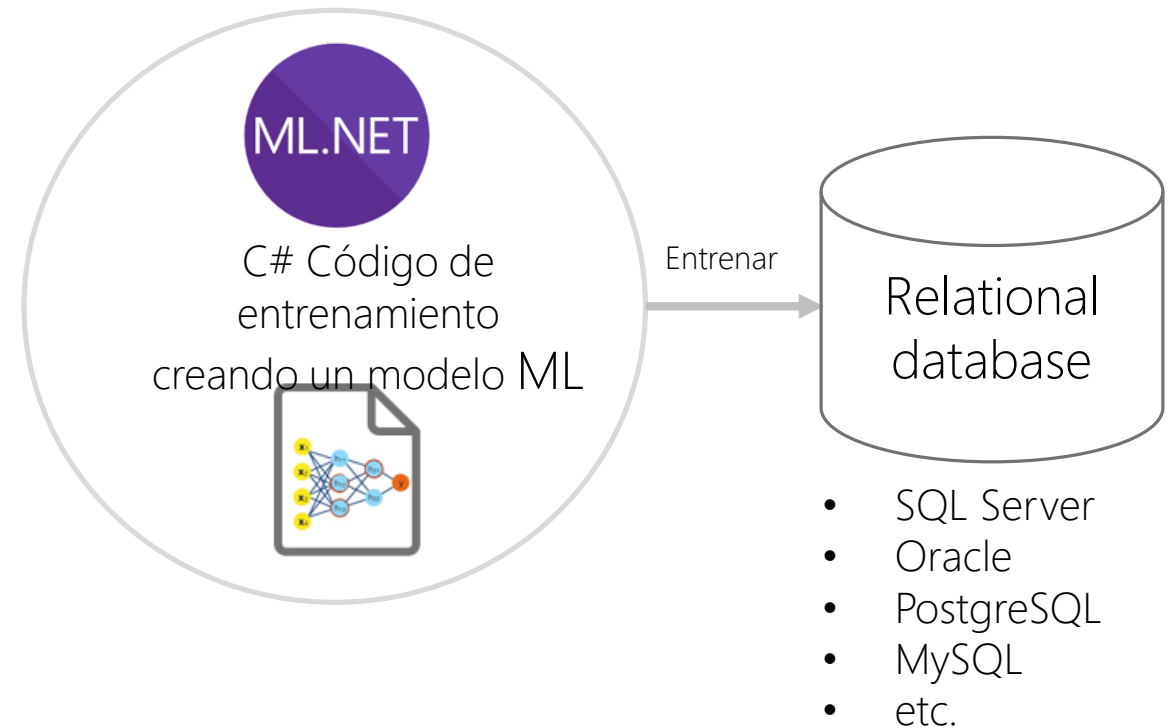




# Lectura de Base de datos

## Escenarios habilitados

- Entrenamiento directamente contra bases de datos relacionales.
- Código simple y listo para usar
- Admite cualquier RDBMS compatible con System.Data
- Actualmente en versión preliminar (versión 1.4-preview)



# Demo 2:

## Cargando Base de datos



FileEditViewProjectBuildDebugArchitectureTestAnalyzeToolsExtensionsWindowHelp

Search (Ctrl+Q)

cnvML3T

Live Share

ML.NET Model BuilderProgram.cs

1. Scenario

2. Data

3. Train

4. Evaluate

5. Code

Add data

In order to build a model, you must add data and choose your column to predict.  
[How do I get sample datasets and learn more?](#)

Input

Choose input data source from either SQL Server or File:

File

Select a file: C:\Users\MiguelValle\Desktop\tax  
Supported file formats: .csv or .tsv.

Column to Predict (Label): payment\_type

Input Columns (Features): 6 of 6 columns selected

Data Preview

10 of 100,001 rows and 7 of 7 columns. (1 Label, 6 Features).

payment_type (Label)	vendor_id	rate_code	passenger_count	trip_time_in_secs	trip_distance	fare_amount
CRD	VTS	1	1	1140	3.75	15.5
CRD	VTS	1	1	480	2.72	10
CSH	VTS	1	1	1680	7.8	26.5
CSH	VTS	1	1	600	4.73	14.5

Solution Explorer

Search Solution Explorer (Ctrl+)

Solution 'cnvML3T' (1 of 1 project)

cnvML3T

Dependencies

Program.cs

Output

Ready

Type here to search

Add to Source Control

6:57 PM 11/22/2019

FileEditViewProjectBuildDebugArchitectureTestAnalyzeToolsExtensionsWindowHelp

Search (Ctrl+Q)

cnvML3T

Live Share

ML.NET Model BuilderProgram.cs

1. Scenario

2. Data

3. Train

4. Evaluate

5. Code

OverviewDetails

ML Task:multiclass-classification

Training Time:7 seconds

Models Explored (Total):2 | [View Top 2 models explored](#)

Overall accuracy:

56.04%

Try your model

Note: Fields below are pre-filled by row 1 of your data.

vendor\_id

rate\_code

passenger\_count

trip\_time\_in\_secs

trip\_distance

fare\_amount

Results

Solution Explorer

Search Solution Explorer (Ctrl+)

Solution 'cnvML3T' (1 of 1 project)

cnvML3T

Dependencies

Program.cs

Solution ExplorerTeam ExplorerNotifications

Output

Ready

Type here to search

Taskbar icons

6:57 PM 11/22/2019

# Que hay de nuevo para **Deep Learning in ML.NET**





# Deep Learning modelos con ML.NET

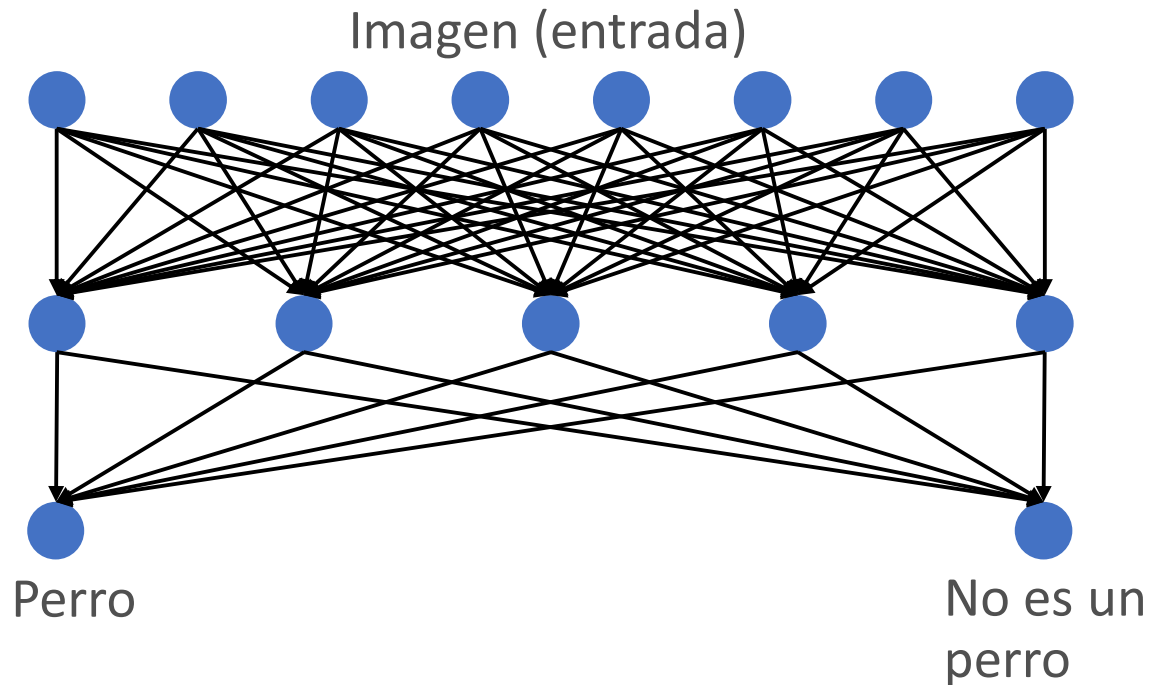
- Agregue inteligencia basada en modelos basados en DNN a sus aplicaciones .NET
- Permite la **visión por computadora** y muchos más dominios de Deep Learning



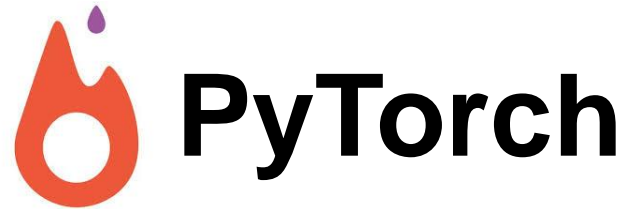
Perro



No es perro



# Bibliotecas líderes en aprendizaje profundo e interoperabilidad



La estrategia de Microsoft es integrar esas bibliotecas de bajo nivel y tiempos de ejecución en ML.NET:

Actualmente compatible con ML.NET:

- TensorFlow
- ONNX

En la hoja de ruta a largo plazo de ML.NET:

- pyTorch

# Liderando el aprendizaje profundo "modelos pre-entrenados" (arquitecturas DNN)

## Computer Vision

### Clasificación Imágenes

- Google InceptionV3
- Microsoft ResNet
- NASNet
- Oxford VGG Model
- MobileNetV2
- etc.

### Detección de Objetos

- Yolo (You Only Look Once)
- R-CNN
- SPP-net
- Fast R-CNN
- Faster R-CNN
- etc.

## Audio y habla

- Wavenet
- espnet
- waveblow
- deepspeech2
- loop
- tacotron
- etc.

**NLP** (Natural language processing)

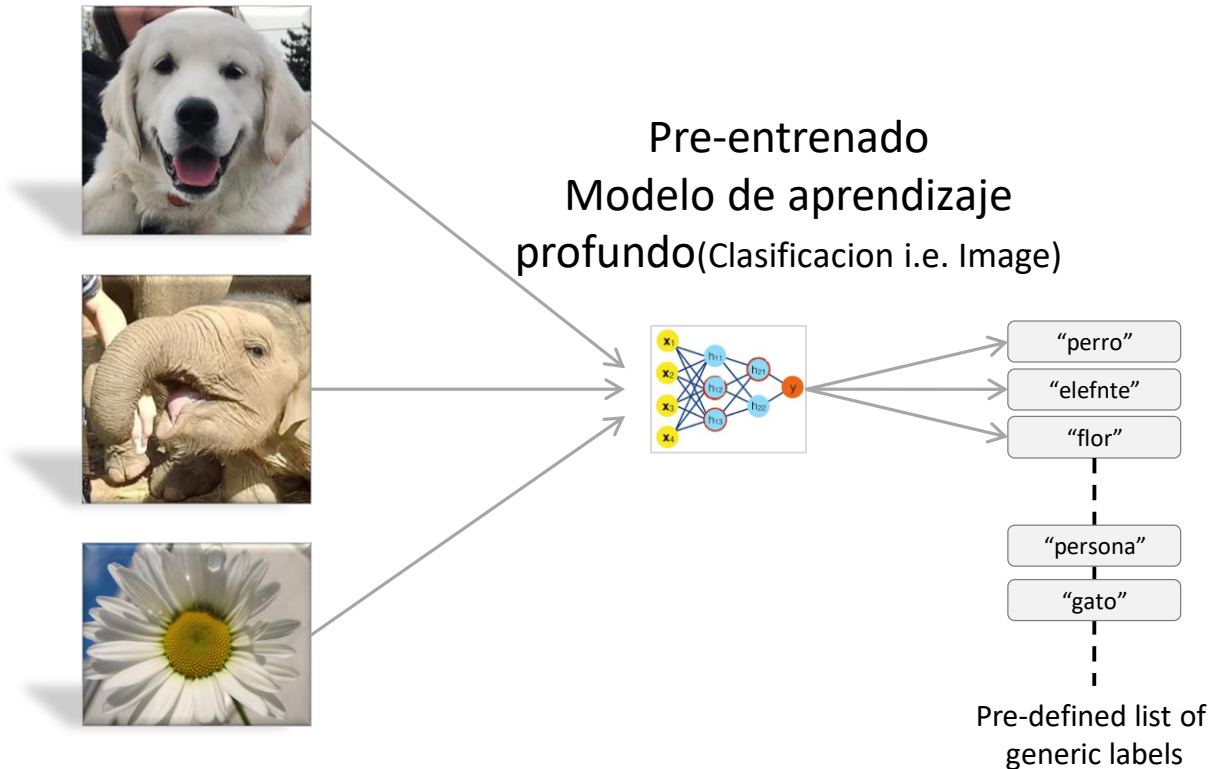
**Modelos generativo**

**... otros dominios...**

- Gran inversión / costo en investigación de arquitectura DNN más capacitación costosa en grandes conjuntos de datos realizados por organizaciones como Google, Microsoft, Facebook, universidades e investigadores.
- Puede aprovecharlo simplemente consumiendo modelos pre-entrenados

# Consumir modelos de entrenamiento deep learning con ML.NET

Escenario: clasificacion imagen (Consumiendo el modelo)



Ejemplo de entrenamiento de modelo (Clasificacion de Imagen):

- Google **Inception v3**, **NASNet**
- Microsoft **ResNet**
- Oxford **VGG** Model, etc.

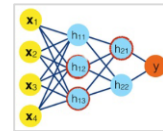
[GitHub sample here](#)

# Consumiendo entrenado deep learning modelo con ML.NET

Escenario: **Deteccion de objetos** (Consumiendo el modelo)

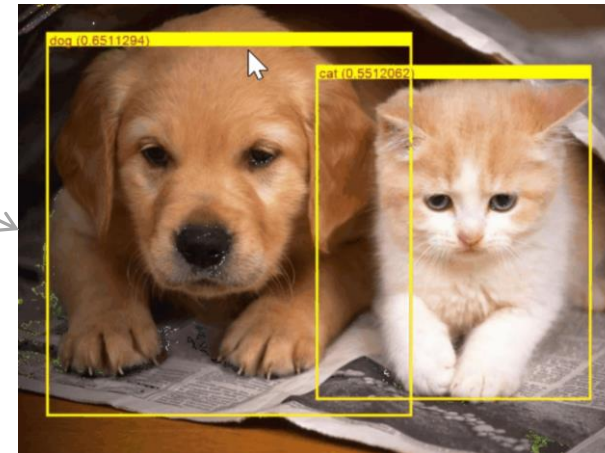


Entrenando  
Modelo Deep learning  
(i.e. Deteccion de Objecto)



Ejemplo de entrenamiento  
Modelo para deteccion objeto:

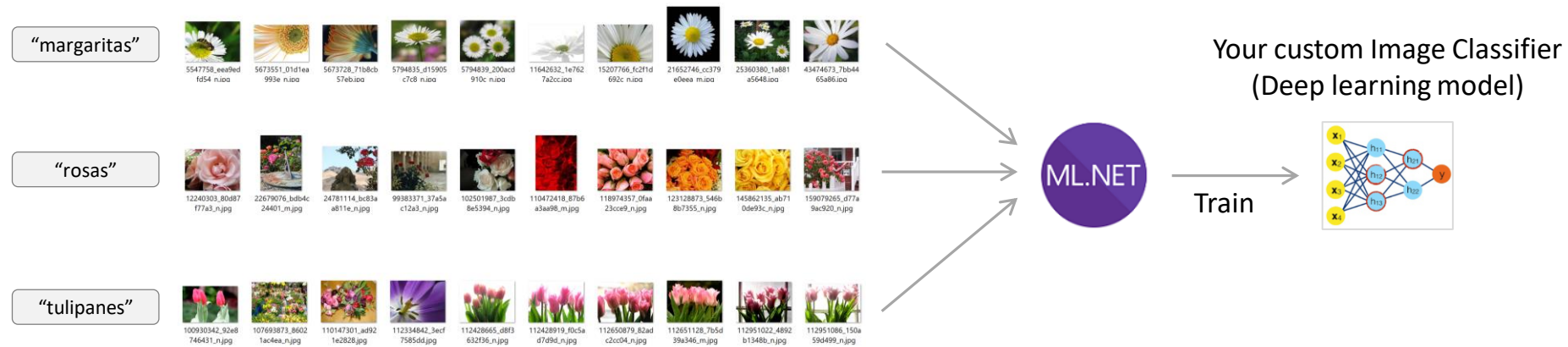
- **YOLO** (Solo miras una vez)
- **Faster R-CNN**
- **SSD**, etc.



# Que sucede si deseo clasificar based on my own custom domain?

Escenario: Clasificacion Imagenes (entrenamiento)

Then you train your own custom deep learning model with ML.NET

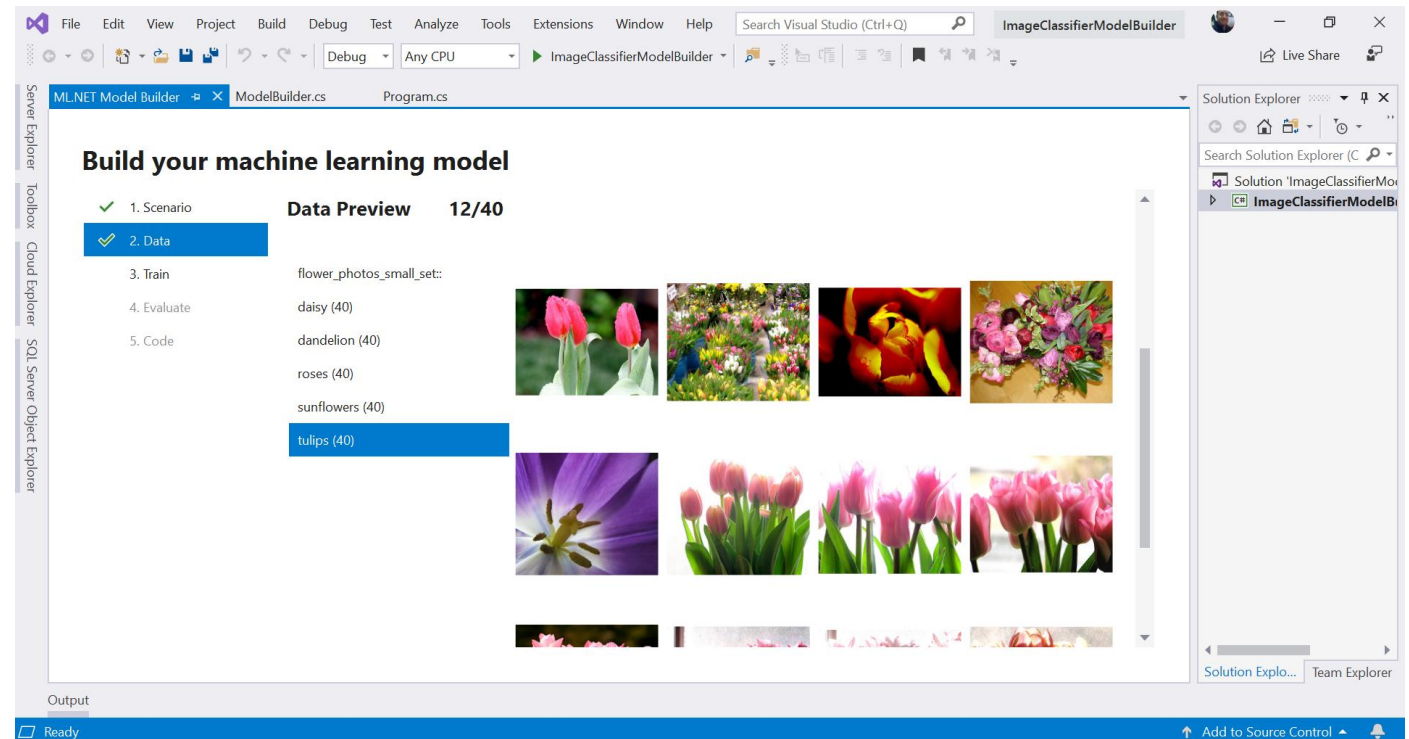


- Al entrenar, ML.NET se basa en TensorFlow debajo (Transfer Learning).
- Actualmente, ML.NET admite la "capacitación de ImageClassifier" (Vista previa)  
El entrenamiento de detección de objetos llegará pronto



# Clasificación de imágenes en ML.NET Model Builder (extensión VS)

- Una interfaz de usuario simple para construir fácilmente modelos ML personalizados con ML automatizado
- Cargar directamente carpetas de imágenes
- Generar código para capacitación y consumo.
- Ejecuta todo local



(\*) La función del clasificador de imágenes en Model Builder será una vista previa pública en los próximos lanzamientos.

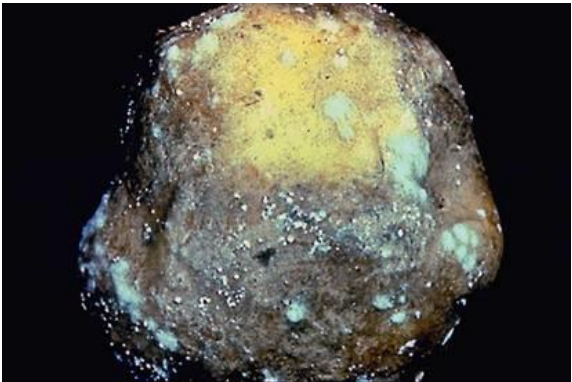
A partir del 23 de septiembre de 2019, esta característica aún no se ha lanzado públicamente en Model Builder. Sin embargo, ya puedes usar la API en la vista previa pública.



# Presentación del problema

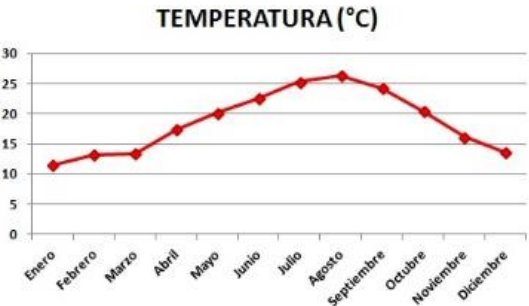
Necesidad de que los agricultores de diferentes zonas agrícolas del Perú cuenten con información para prevenir la enfermedad del hielo.

Región	Superficie cosechada (ha)			Producción (t)			Rendimiento (t/ha)			Precio al productor (S/ / t)		
	Oct	Nov	Dic	Oct	Nov	Dic	Oct	Nov	Dic	Oct	Nov	Dic
NACIONAL	10 955	14 617	15 745	221 569	283 320	270 708	20,2	19,4	17,2	538	507	612
Amazonas	317	133	94	5 015	2 056	1 495	15,8	15,5	15,9	659	808	681
Ancash	108	215	395	1 249	2 333	4 188	11,6	10,8	10,6	809	750	697
Apurimac	59	44	353	1 222	759	5 823	20,7	17,3	16,5	471	436	428
Arequipa	1 475	1 707	886	53 057	58 414	30 443	36,0	34,2	34,4	593	388	359
Ayacucho	23	75	223	401	1 192	4 121	17,4	15,9	18,5	590	601	404
Cajamarca	2 039	2 526	3 319	23 713	28 935	35 296	11,6	11,5	10,6	869	679	726
Cusco	6	53	232	90	525	3 004	15,0	15,9	12,9	800	700	727
Huanuavelica	612	660	678	6 911	8 520	6 780	11,3	12,9	10,0	480	306	431
Huánuco	3 972	4 947	4 323	75 967	93 240	84 400	19,1	18,8	19,5	406	352	433
Ica	585	158	0	19 460	5 104	0	33,3	32,3	-	463	532	-
Junín	72	601	1 059	1 229	10 012	18 796	17,1	16,7	17,7	435	384	316
La Libertad	946	1 795	1 404	19 141	35 792	28 972	20,2	19,9	20,6	592	695	752
Lambayeque	0	0	150	0	0	1 200	-	8,0	-	-	-	500
Lima (excluye LM)	444	858	680	10 854	23 451	17 081	24,4	27,3	25,1	415	285	241
Lima Metropolitana	22	9	0	631	222	0	28,7	24,7	-	320	557	-
Moquegua	3	0	4	83	0	52	27,7	-	13,0	750	-	630
Passco	2	58	197	42	1 163	3 921	21,0	20,1	19,9	500	307	303
Piura	262	29	233	2 358	191	2 223	9,0	6,6	9,5	857	837	647
Puno	0	760	1 509	0	11 227	22 792	-	14,8	15,1	-	1 974	1 937
Tacna	8	10	7	147	183	121	18,4	18,3	17,3	900	900	831



Solanum tuberosum

Phytophthora infestans



Modelo Analítico Predictivo:



# Mapa de Estrategia y Tecnología

Estrategia		Tomar decisiones acertadas gracias al uso del big data, IOT y aplicación de IA para poder tratar las plantas de la papa en el proceso de crecimiento, basado en ciertos parámetros permitiendo enviar mensajes de alerta a los agricultores									
Objetivo		Desarrollar un plan de negocios que demuestre la viabilidad económica del proyecto para predios agrícolas, mejorando el tiempo de respuesta a determinadas condiciones climatológicas	Diseñar un sistema que permita la interacción entre un dispositivos electrónicos instalados en una áreas agrícolas para monitorear y controlar diferentes parámetros climáticos.	Desarrollar un módulo de análisis predictivo de comportamientos climático, que permita el análisis de la información de parámetros climáticos recolectados	Desarrollar un módulo de toma de decisiones y alertas a traves de comunicaciones web y moviles, para la prevención de enfermedades agrícolas.		...	...	...	...	...
Caso de Uso		Entender la geolocalización de zonas de cultivo	Muestreo y datos requeridos	Análisis predictivo	Pruebas e Implementación con sensores IOT	Entrenamiento/Aprendizaje	...	...	...	...	...
Necesidad		Extraer datos GIS, almacenarlos, analizarlos, visualizarlos	Extraer datos transaccionales, de los sensores y fuentes de datos del tiempo NASA, SENAMI	Aplica metodología y algoritmos de Minería de datos como serie de tiempo	Desarrolla las interfaces del software y la comunicación los sensores	Desarrolla pruebas para calibrar el modelo	...	...	...	...	...
TECNOLOGIA	Machine Learning e IA		X	X	X	X					
	GIS	X	X								
	Big Data & Analítica	X	X	X	X	X					
	IoT	X	X	X	X	X					
	ChatBot										
	IVR										
	cloud Computing		X	X	X	X					
	App Movil	X				X					
	Robótica										

## Sensores

## Variables climáticas

## Variables

## Categoría de variable

## Unidad de medida

## Tipo unidad de medida

## Vegetales

## Plagas

## Monitoreo

## Seguridad

## Variables climáticas

Buscar: [Nueva variable](#)

ID	Abreviatura	Nombre de la variable	Categoría	UM	Tipo UM	Editar
031	UVDOSE	Dosis UV	radiación solar	índice UV	indicador	<a href="#">Editar</a>
029	EMC	Equilibrio de contenido de humedad	humedad	índice EMC	indicador	<a href="#">Editar</a>
028	ET	Evotranspiración	vapor	milímetro cúbico (mm <sup>3</sup> )	volumen	<a href="#">Editar</a>
027	RAINRATE	Intensidad de lluvia	lluvia	milímetros cúbicos por hora (mm <sup>3</sup> /h)	velocidad	<a href="#">Editar</a>
026	RAIN	Lluvia	lluvia	milímetro cúbico (mm <sup>3</sup> )	volumen	<a href="#">Editar</a>
025	RADSOLMAX	Radiación solar máxima	radiación solar	watts por metro cuadrado (W/m <sup>2</sup> )	irradiancia	<a href="#">Editar</a>
024	RADSOL	Radiación solar	radiación solar	watts por metro cuadrado (W/m <sup>2</sup> )	irradiancia	<a href="#">Editar</a>
023	UVMAX	Índice UV máxima	radiación solar	índice UV	indicador	<a href="#">Editar</a>
022	UV	Índice UV	radiación solar	índice UV	indicador	<a href="#">Editar</a>
021	ENERGSOL	Energía solar	radiación solar	langley (Ly)	insolación	<a href="#">Editar</a>
020	PRESBAR	Presión barométrica	presión atmosférica	milibars (mb)	presión	<a href="#">Editar</a>
019	AIRDENS	Densidad del aire	presión atmosférica	kilogramo por metro cúbico (kg/m <sup>3</sup> )	densidad	<a href="#">Editar</a>
018	WINDCHILL	Sensación térmica	calor	grados centigrados (°C)	temperatura	<a href="#">Editar</a>
017	THSW	Índice THSW	calor	grados centigrados (°C)	temperatura	<a href="#">Editar</a>

Sensores

Variables climáticas

Variables

Categoría de variable

Unidad de medida

Tipo unidad de medida

Vegetales

Plagas

Monitoreo

Seguridad

## Editar variable

Nombre:

Velocidad del viento

Abreviatura:

WINDSPD

Categoría:

viento

U.M.:

kilómetros por hora (km/h)

Tipo cálculo:

Promedio

Descripción:

Promedio de todas las velocidades del viento calculadas para cada intervalo de tiempo desde el modelo anterior.

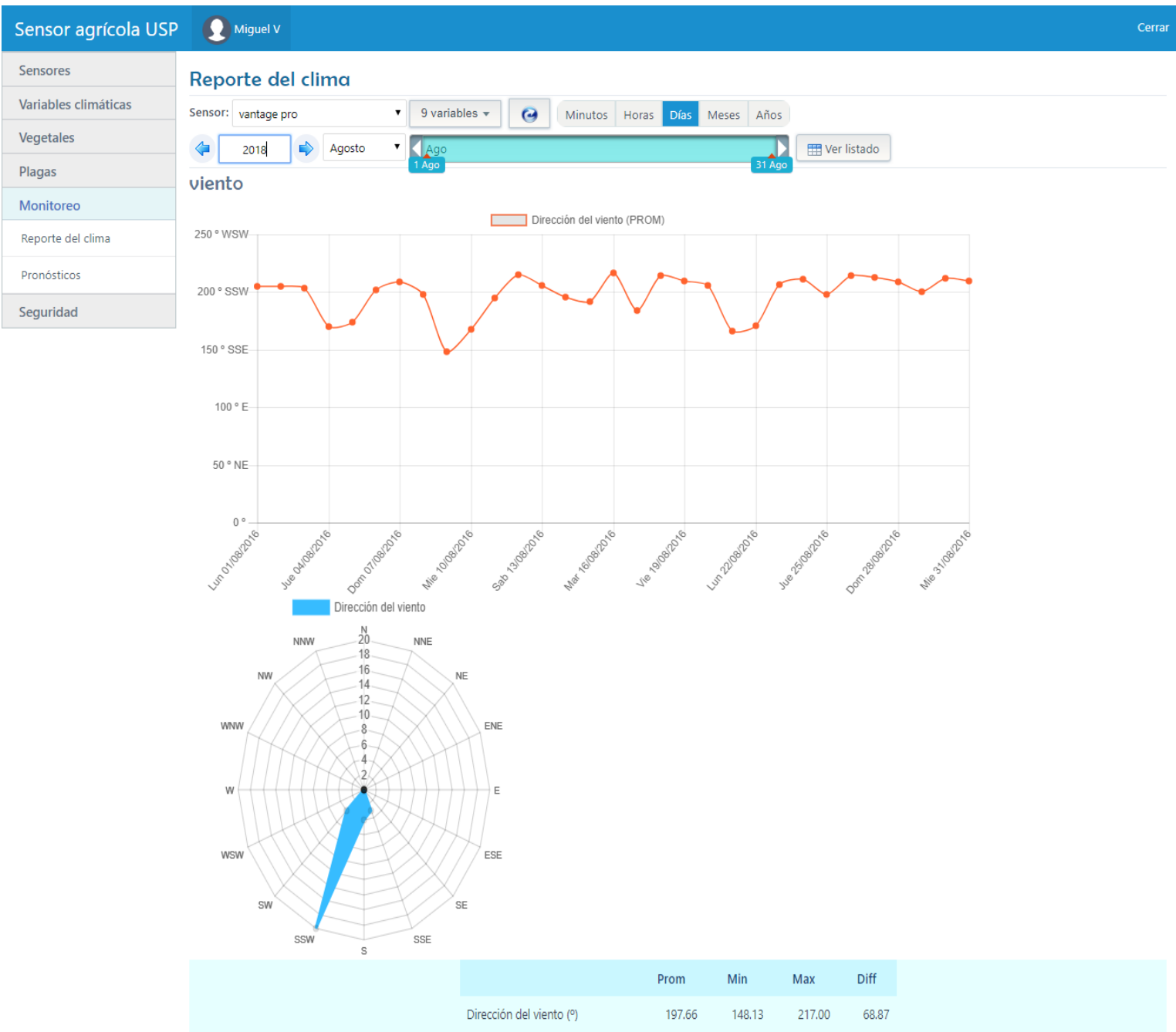
## Escala de variable

 Agregar escala

Nº	Escala	Valor min	Valor max	U.M.	Color	Alertar	Quitar
1	Calma	0	2	km/h		Alertar	Quitar
2	Ventolina	2	6	km/h		Alertar	Quitar
3	Brisa muy débil	7	11	km/h		Alertar	Quitar
4	Brisa débil	11	20	km/h		Alertar	Quitar
5	Brisa moderada	20	29	km/h		Alertar	Quitar
6	Brisa fresca	30	39	km/h		Alertar	Quitar
7	Brisa fuerte	40	50	km/h		Alertar	Quitar
8	Viento fuerte	50	61	km/h		Alertar	Quitar
9	Temporal	62	74	km/h		Alertar	Quitar
10	Temporal fuerte	75	87	km/h		Alertar	Quitar
11	Temporal duro	88	101	km/h		Alertar	Quitar
12	Temporal muy duro	102	117	km/h		Alertar	Quitar
13	Huracán	117	1000	km/h		Alertar	Quitar

Guardar

Cancelar



Outlook

Buscar

S<sup>2</sup>

⚙

?

+

Mensaje nuevo

🗑

Eliminar

📁

Archivo

🚫

No deseado

🧹

Limpiar

📁

Mover a

🏷

Categorizar

...

🔍

Versión beta de Outlook

▼ Favoritos

📁 Archivo

📁 Elementos enviados

📁 Bandeja de en...

4874

Agregar favorito

▼ Carpetas

📁 Bandeja de en...

4874

✉

🚫

Correo no desea... 216

📅

✎

Borradores 64

📁 Elementos enviados

🗑 Elementos eliminad...

Prioritarios

Otro

Filtrar ▼

10

Otros: nuevas conversaciones

Pierre Filipe Diogo Rebatta a través de Linke...

○

Estacion climatica

Alerta climatica

Vie 10:22 PM

Los niveles de velocidad del viento han alcanz...

OP

OLGA MANOSALVA PALMA

FORO PARA CARNE ÚNICO MA...

Vie 08:35 PM

Estimados Maestristas, La presente para salud...

TM

Tom Makowski

RevDeBug licence keys, Global ...

Vie 03:21 AM

Hello Miguel! In the attachment I send five 1 y...

📎

RevDeBug\_GAB2018Per...

69 KB

+4

Alerta climatica

EC

Estacion climatica <syscontrolpro@h  
otmail.com>

Vie 20/04/2018, 10:22 PM

Estacion climatica; ▼

Los niveles de **velocidad del viento** han alcanzado la categoría  
**temporal fuerte** con un valor de **75.00 km/h**, el día  
**20/04/2018** a las 07:36 pm en la parcela Cultivos Agronomía

Responder

CARTERAS  
& ACCESORIOS

falabella.com

LOS QUIERO TODOS

# ML.NET recursos



Get started at <http://dot.net/ml>



Try the samples at <http://aka.ms/mlnetsamples>



Read the docs at <http://aka.ms/mlnetdocs>



Watch ML.NET videos at <https://aka.ms/mlnetyoutube>



Request features or contribute at <http://aka.ms/mlnet>



# Thank you! 😊

Miguel Arturo Valle Pelaez  
Microsoft MVP IA  
@mavpelaez

