**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**Phân hiệu tại TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----🙣🕮🙡----**

**BÁO CÁO TIỂU LUẬN**

**ĐỀ TÀI:**

**TÌM HIỂU CÔNG CỤ MICROSOFT ML.NET**

**Môn:** CÔNG NGHỆ PHẦN MỀM

**Sinh viên thực hiện: 1.** VÕ TẤN LỄ

**2.** NGUYỄN THANH SANG

**3.** TRẦN LÊ QUYỀN

**4.** ĐẶNG QUANG TRƯỜNG NGUYÊN

**5.** NGUYỄN VĂN DU

**Lớp:** CÔNG NGHỆ THÔNG TIN

**Giảng viên hướng dẫn:** Gv. TRẦN PHONG NHÃ

*TP Hồ Chí Minh, năm 2021*

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**Phân hiệu tại TP. HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----🙣🕮🙡----**

**BÁO CÁO TIỂU LUẬN**

**ĐỀ TÀI:**

**TÌM HIỂU CÔNG CỤ MICROSOFT ML.NET**

**Môn:** CÔNG NGHỆ PHẦN MỀM

**Sinh viên thực hiện: 1.** VÕ TẤN LỄ

**2.** NGUYỄN THANH SANG

**3.** TRẦN LÊ QUYỀN

**4.** ĐẶNG QUANG TRƯỜNG NGUYÊN

**5.** NGUYỄN VĂN DU

**Lớp:** CÔNG NGHỆ THÔNG TIN

**Giảng viên hướng dẫn:** Gv. TRẦN PHONG NHÃ

*TP Hồ Chí Minh, năm 2021*

|  |  |
| --- | --- |
| **TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  **PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc** |

# **NHIỆM VỤ BÁO CÁO TIỂU LUẬN**

BỘ MÔN: **CÔNG NGHỆ THÔNG TIN**

-------\*\*\*-------

**Họ tên SV: Mã sinh viên:**

1. VÕ TẤN LỄ **1.** 5951071048
2. NGUYỄN THANH SANG **2.** 5951071090
3. TRẦN LÊ QUYỀN **3.** 5951071088
4. ĐẶNG QUANG TRƯỜNG NGUYÊN **4.** 6051071078
5. NGUYỄN VĂN DU  **5.** 6051071016

**Lớp:** Công Nghệ Thông Tin

1. **Tên đề tài:**

TÌM HIỂU CÔNG CỤ MICROSOFT ML.NET

1. **Mục đích, yêu cầu:**
   1. **Mục đích:**

Tiếp cận công nghệ Trí tuệ nhân tạo, Machine Learning, tìm hiểu về các công cụ AI cơ bản.

Áp dụng công cụ vào thực tiển, xây dựng chương trình cơ bản dựa trên công cụ tìm hiểu.

* 1. **Yêu cầu:**
     + Tìm hiểu được các khái niệm và cách hoạt động về thư viện Microsoft ML.Net.
     + Các ứng dụng và sự phát triển của Microsoft ML.Net.
     + Xây dựng được 1 chương trình cơ bản về công cụ đó.

1. **Nội dung hướng đến**

* Machine Learning là gì? Các khái niệm về nó.
* Tổng quan về Microsoft ML.Net.
* Nền tảng xây dựng.
* Kiểm thử và xây dựng chương trình.
* Kết quả thu được

|  |  |
| --- | --- |
|  | *Ngày …… tháng …… năm 2021*  **Giảng viên hướng dẫn**  *(Ký và ghi họ và tên)*  ………………………………………………. |

# **LỜI CẢM ƠN**

Lời đầu tiên nhóm em xin gửi lời cảm ơn chân thành đến quý thầy **TRẦN PHONG NHÃ** trong **Bộ môn Công nghệ thông tin – Phân hiệu Trường Đại học Giao thông vận tải**, thầy đã dạy nhóm em môn **CÔNG NGHỆ PHẦN MỀM.** Thầy đã truyền dạy, đã trang bị cho em kho tàng kiến thức về bầu trời côngnghệ thông tin rộng lớn.

Ở đây, em không chỉ học được kiến thức về sách vở mà nhóm em còn học được các bài học, kỹ năng sống trước khi tạm biệt mái trường đại học thân yêu này và tiến ra biển đời mênh mông rộng lớn. Đặc biệt, em xin gửi lời cảm ơn chân thành và sâu sắc đến thầy**.**

Trong quá trình học tập và tìm hiểu em đã nỗ lực rất nhiều với mong muốn hoàn thành project một cách tốt nhất, nhưng đời người sẽ có những thiếu sót không thể tránh khỏi, và với những người chưa chững chạc và trưởng thành như em thì sai lầm là không thể không mắc phải. Em mong thầy, cô bộ môn có thể thông cảm và cho em những ý kiến, đóng góp để em có thể hoàn thành tốt hơn về BÀI TÌM HIỂU của mình một cách trọn vẹn để có cơ hội sửa chữa.

Sau cùng, em xin kính chúc Thầy, lời chúc sức khoẻ, an toàn luôn hạnh phúc và thành công hơn nữa trong công việc cũng nhưtrong cuộc sống.

Em xin chân thành cảm ơn!

# 

# **NHẬN XÉT CỦA GIÁO VIÊN**

|  |  |
| --- | --- |
|  | *Tp. Hồ Chí Minh, ngày …… tháng …… năm 2021*  **Giảng viên hướng dẫn**  *(Ký và ghi họ và tên)*  ………………………………………………. |

MỤC LỤC

[NHIỆM VỤ BÁO CÁO TIỂU LUẬN 3](#_Toc83929300)

[LỜI CẢM ƠN 5](#_Toc83929301)

[NHẬN XÉT CỦA GIÁO VIÊN 6](#_Toc83929302)

[MỞ ĐẦU 9](#_Toc83929303)

[1. Lý do chọn đề tài. 9](#_Toc83929304)

[2. Mục tiêu nghiên cứu: 9](#_Toc83929305)

[3. Phương pháp nghiên cứu 9](#_Toc83929306)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VỀ MICROSOFT ML.NET 10](#_Toc83929307)

[1.1. Machine Learing (Máy học). 10](#_Toc83929308)

[1.1.1. Khái niệm. 10](#_Toc83929309)

[1.1.2. Phân loại thuật toán trong Machine Learning 11](#_Toc83929310)

[1.1.3. Các loại thuật toán và hướng sử dụng. 12](#_Toc83929311)

[1.1.4. Machine Learning Workflow 20](#_Toc83929312)

[1.1.5. Ứng dụng của Machine Learning 22](#_Toc83929313)

[1.2. ML.Net 22](#_Toc83929314)

[**1.2.1.** **Khái niệm .Net** 22](#_Toc83929315)

[**1.2.2.** **Khái niệm ML.Net** 23](#_Toc83929316)

[**1.2.3.** **Các công cụ hổ trợ trong ML.Net** 24](#_Toc83929317)

[**1.2.4.** **Các ứng dụng của ML.Net** 24](#_Toc83929318)

[**1.2.5.** **Cách cài đặt thư viện ML.Net** 25](#_Toc83929319)

[CHƯƠNG 2: DEMO CHƯƠNG TRÌNH” NHẬN DIỆN” 31](#_Toc83929320)

[2.1. Chương trình “ Nhận diện khuôn mặt” 31](#_Toc83929321)

[2.2. Chương trình nhận diện biển số xe 32](#_Toc83929322)

[CHƯƠNG 3: KẾT LUẬN 34](#_Toc83929323)

[TÀI LIỆU THAM KHẢO 35](#_Toc83929324)

**DANH MỤC HÌNH ẢNH**

[Hình 1: Lịch sử Machine Learning 10](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270715)

[Hình 2: Cơ chế hoạt động 11](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270716)

[Hình 3: Thuật toán Cây quyết định 13](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270717)

[Hình 4: Thuật toán K-means 13](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270718)

[Hình 5: Thuật toán Mạng nơ-ron nhân tạo 14](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270719)

[Hình 6: Thuật toán Học tăng cường 14](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270720)

[Hình 7: Thuật toán phân loại tự động Bayes 15](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270721)

[Hình 8: Thuật toán quy hồi tuyến tính 16](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270722)

[Hình 9: Thuật toán Quy hồi Logictics 17](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270723)

[Hình 10: Thuật toán SVM 17](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270724)

[Hình 11: Thuật toán kết hợp nhiều phương pháp 18](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270725)

[Hình 12: Thuật toán phân tích dữ liệu phần kết lõi 19](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270726)

[Hình 13: Thuật toán phân tích dữ liệu độc lập 20](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270727)

[Hình 14: Machine Learning WorkFlow 21](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270728)

[Hình 15: ML.Net 23](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270729)

[Hình 16: Tạo Project 25](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270730)

[Hình 17: Chọn dự án 26](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270731)

[Hình 18: Cài đặt gói NuGet 26](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270732)

[Hình 19: Add Machine Learning 27](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270733)

[Hình 20: Chọn loại ứng dụng 27](#_Toc84270734)

[Hình 21: Add File cho Máy học 28](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270735)

[Hình 22: Chọn loại nền tảng để Máy học 28](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270736)

[Hình 23: Máy tiến hành Training 29](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270737)

[Hình 24: Kết quả Training 29](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270738)

[Hình 25: Tiến hành Code 30](file:///C:\Users\dell\Downloads\Trí%20tuệ%20nhân%20tạo\Bài%20tiểu%20luận%20Trí%20tuệ%20nhân%20tao(AI).docx#_Toc84270739)

[Hình 26: Nhận diện chữ viết tay 31](#_Toc84270740)

# **MỞ ĐẦU**

1. **Lý do chọn đề tài.**

Trong dòng chảy của cuộc cách mạng công nghiệp 4.0, trí tuệ nhân tạo và Machine Learning ngày càng phổ biến hơn trong mọi lĩnh vực của cuộc sống. Dù được John McCarthy – nhà khoa học máy tính người Mỹ đề cập lần đầu tiên vào những năm 1950 nhưng đến ngày nay thuật ngữ trí tuệ nhân tạo mới thực sự được biết đến rộng rãi và được các “ông lớn” của làng công nghệ chạy đua phát triển. Có thể kể đến như Google, Apple, Facebook, Microsoft, IBM…

Trí tuệ nhân tạo còn được biết đến với cái tên trí thông minh nhân tạo (Artificial Intelligence – viết tắt là AI), một lĩnh vực thuộc ngành khoa học máy tính và công nghệ thông tin. Đây là trí tuệ do con người lập trình với mục tiêu giúp máy tính có thể tự động hóa các hành vi một cách thông minh như con người. Trí tuệ nhân tạo có khả năng tự học hỏi, tự phát triển, tự lập luận giải quyết vấn đề, biết giao tiếp do hiểu tiếng nói, biết tự thích nghi thông qua các dữ liệu được nạp vào và tái lập trình với những kiến thức mới.

Machine Learning cũng như là Trí tuệ nhân tạo được xây dựng và hổ trợ bởi rất nhiều công cụ.Ví dụ: TensorFlow, Soar, Jade, H2o…Nhóm em nhận thấy khả năng nhận diện bằng công cụ mã nguồn mở Microsoft ML.Net khá thú vị và khả năng ứng dụng vào thực tế vô cùng thiết thực. Đó là lý do nhóm em chọn tìm hiểu về thư viện ML.Net.

1. **Mục tiêu nghiên cứu:**

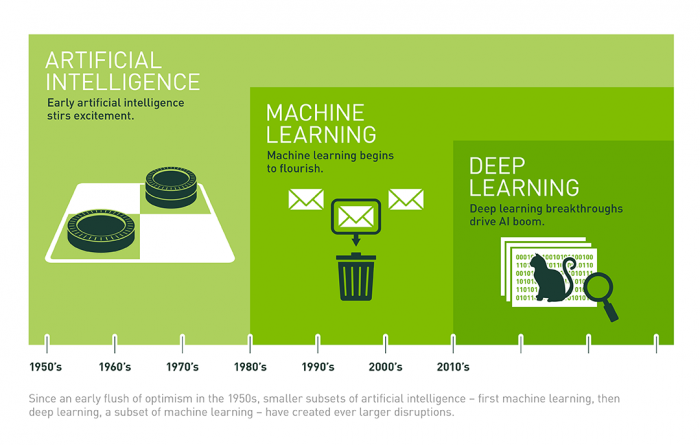
* Tìm hiểu và nghiên cứu về các khái niệm về Machine Learning của Microsoft ML.Net
* Phân tích và xây dựng chương trình minh hoạ đơn giản về công cụ.

1. **Phương pháp nghiên cứu**

* Lý thuyết: Tìm hiểu sơ lược Machine Learning, nền tảng xây dựng, các hổ trợ của ML.Net.
* Thực hành: Tiến hành xây dựng chương trình.

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT VỀ MICROSOFT ML.NET**

* 1. **Machine Learing (Máy học).**
     1. **Khái niệm.**

Machine Learning là một tập con của AI. Theo định nghĩa của Wikipedia, Machine learning is the subfield of computer science that “gives computers the ability to learn without being explicitly programmed”. Nói đơn giản, Machine Learning là một lĩnh vực nhỏ của Khoa Học Máy Tính, nó có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình cụ thể.

Hình 1: Lịch sử Machine Learning

Machine learning (ML hay học máy) là một phạm trù trong thuật toán, cho phép các ứng dụng phần mềm dự đoán kết quả chính xác hơn mà không cần lập trình trực tiếp. Tiên đề của machine learning là xây dựng thuật toán có thể nhận thông tin và sử dụng các phân tích thống kê để dự đoán kết quả, đồng thời cập nhật kết quả khi có dữ liệu mới.

Quá trình trong machine learning cũng giống như quá trình khai phá dữ liệu (data mining) và mô hình dự báo. Cả hai đều cần phải có dữ liệu chi tiết để thay đổi hoạt động chương trình một cách hợp lý. Nhiều người đã có thể làm quen với machine learning thông qua mua sắm trực tuyến và được đề xuất các quảng cáo liên quan đến việc mua sắm của họ. Điều này là bởi vì các công cụ đề xuất sử dụng machine learning để cá nhân hóa quảng cáo trong hầu hết thời gian thực. Ngoài ra, còn có thể sử dụng machine learning để phát hiện lỗi, lọc thư rác, phát hiện lỗ hổng an ninh mạng, bảo trì, và xây dựng kho tin tức.

Hình 2: Cơ chế hoạt động

Đối với máy học, bạn sẽ **cần phải có dữ liệu** để huấn luyện mô hình, đây chính là quá trình mà mô hình máy học tìm ra các quy tắc như đã nói bên trên. Sau khi huấn luyện xong, máy tính sẽ dựa vào mô hình đã huấn luyện để đưa ra **dự đoán**.

* + 1. **Phân loại thuật toán trong Machine Learning**

Thuật toán trong machine learning dựa vào **cách máy tính học** được chia thành:

* **Thuật toán giám sát (Supervised learning)** đòi hỏi phải có một chuyên gia dữ liệu với kỹ năng machine learning để cung cấp cả dữ liệu đầu vào và đầu ra, đồng thời đưa ra những nhận xét về tính chính xác của các dự đoán trong quá trình đào tạo. Chuyên gia phân tích dữ liệu sẽ quyết định các nhân tố, đặc tính, hay mô hình nào sẽ được phân tích và sử dụng để phát triển dự đoán. Khi kết thúc đào tạo, thuật toán sẽ áp dụng những gì học được với dữ liệu mới.
* **Thuật toán không giám sát (Unsupervised learning )** không cần được đào tạo với dữ liệu đầu ra mong muốn. Thay vào đó, nó sử dụng một phương pháp lặp được gọi là "**deep learning**" để phân tích dữ liệu và đưa ra kết luận. Thuật toán không giám sát (hay **mạng nơ-ron nhân tạo - "neural networks"**) được sử dụng trong những công việc phức tạp hơn, bao gồm nhận diện hình ảnh, chuyển lời nói thành văn bản, và xử lý ngôn ngữ tự nhiên. Những mạng nơ-ron nhân tạo này sẽ kết hợp hàng triệu ví dụ để tự động nhận diện mối tương quan giữa các nhân tố. Sau khi được đào tạo, thuật toán sẽ sử dụng ngân hàng liên kết để phân tích dữ liệu mới. Những thuật toán này chỉ khả thi trong thời đại "big data," vì nó đòi hỏi một lượng dữ liệu bài giảng khổng lồ.
* **Thuật toán bán giám sát (Supervised learning)** Các thuật toán học máy được giám sát bán nằm ở đâu đó giữa học tập có giám sát và không giám sát, vì chúng sử dụng cả dữ liệu được gắn nhãn và không nhãn cho đào tạo - thường là một lượng nhỏ dữ liệu được gắn nhãn và một lượng lớn dữ liệu không được gắn nhãn. Các hệ thống sử dụng phương pháp này có thể cải thiện đáng kể độ chính xác trong học tập.
* **Thuật toán học máy gia cố (Reinforced learning)** Các thuật toán học máy gia cố là một phương pháp học tương tác với môi trường của nó bằng cách tạo ra các hành động và phát hiện ra các lỗi hoặc manh mối. Thử nghiệm và tìm kiếm lỗi và manh mối Phương pháp này cho phép máy móc, máy tính với phần mềm tự động xác định hành vi lý tưởng trong một bối cảnh cụ thể để tối đa hóa hiệu suất của nó.

Thuật toán trong machine learning dựa vào **chức năng của thuật toán** được chia thành:

* **Regression – Hồi quy**

Hồi quy là những thuật toán có chức năng học và **dự đoán giá trị** ( **thường là giá trị thực – rời rạc**). Ví dụ như bạn đưa các đặc điểm về căn nhà như diện tích sàn, số phòng ngủ… và giá căn nhà cho máy học, sau đó bạn có thể dùng mô hình đã huấn luyện để dự đoán giá các căn nhà – trong đó giá các căn nhà là roi.

* **Classification – Phân lớp**

Phân lớp chính là thuật toán áp dụng trong bài phân loại táo, cam mà lúc đầu mình ví dụ. Loại này thì nhận vào dữ liệu và lớp mà nó thuộc vào, sau khi được huấn luyện thì mô hình có thể **phân lớp các dữ liệu**. Ví dụ như bài táo, cam thì có 2 lớp thôi, chúng ta có thể gọi là binary classificatio. Nếu có nhiều lớp hơn, ví dụ có thể thêm quýt… thì có thể gọi là Multi-Classification.

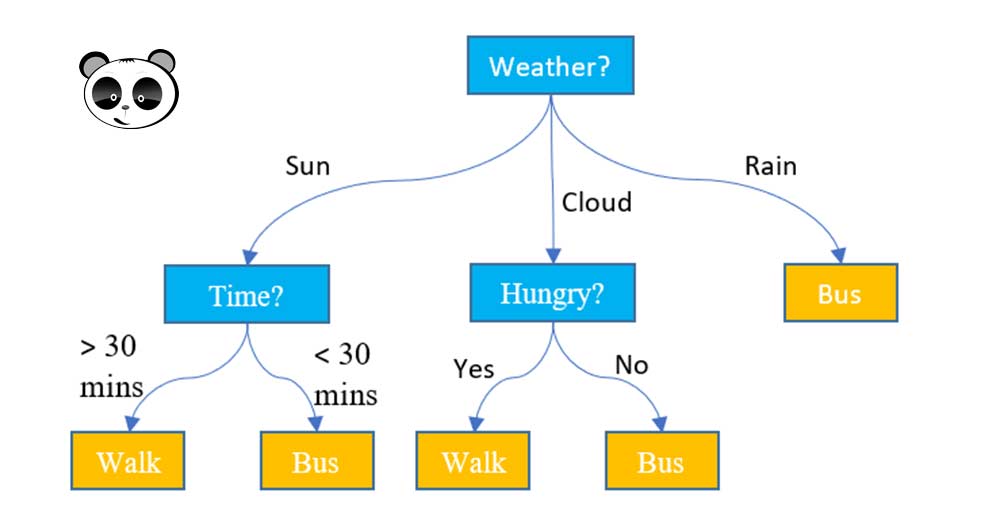
* **Clustering – Phân cụm**

Phân cụm là thuật toán có chức năng **phân cụm dữ liệu**. Thuật toán này thường được áp dụng cho dữ liệu chưa được gán nhãn. Sự phân bố, cấu trúc của dữ liệu phải tự mô hình khám phá và phân cụm chúng, bạn có thể xem hình ảnh bên dưới để hiểu rõ hơn. Hình bên dưới sử dụng thuật toán K-means.

* + 1. **Các loại thuật toán và hướng sử dụng.**

Không có giới hạn nào trong việc sử dụng cũng như thuật toán Machine Learning. Dưới đây là một số mô hình phổ biến nhất:

* **Cây quyết định (Decision Tree):** Mô hình này sẽ quan sát những hành động nhất định để tìm ra kết quả mong muốn

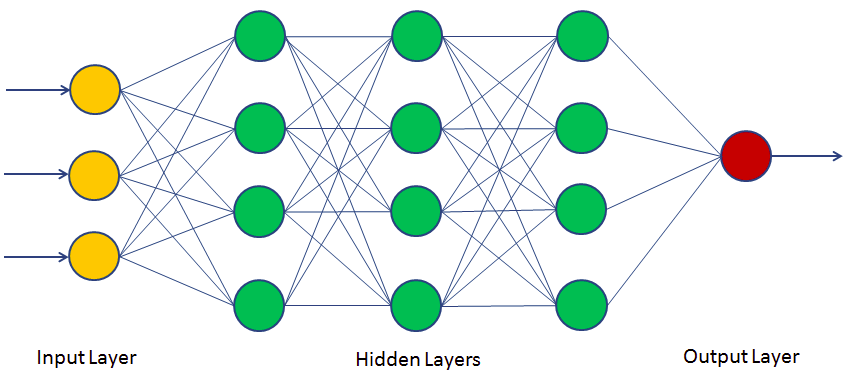


Hình 3: Thuật toán Cây quyết định

* **Phân cụm K-means (K-means clustering):** Mô hình này chia một số điểm dữ liệu nhất định thành các nhóm dựa theo đặc điểm riêng của chúng.

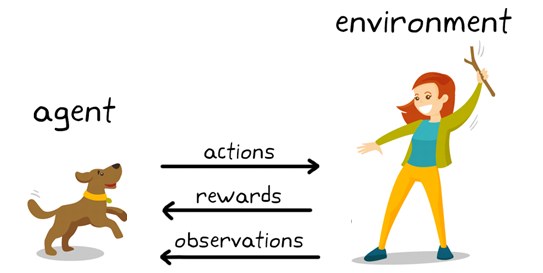
Hình 4: Thuật toán K-means

* **Mạng nơ-ron nhân tạo (Neural networks):** Mô hình này sử dụng một lượng lớn dữ liệu bài giảng để xác định mỗi tương quan giữa các nhân tố nhằm mục đích xử lý dữ liệu mới trong tương lai



Hình 5: Thuật toán Mạng nơ-ron nhân tạo

* **Học tăng cường (Reinforcement Learning):** Lĩnh vực này bao gồm các mô hình giúp hoàn thành một quá trình, khuyến khích các bước đưa ra kết quả thuận lợi và ngăn chặn các bước cho ra kết quả không mong muốn cho đến khi thuật toán đưa ra được quy trình tối ưu nhất

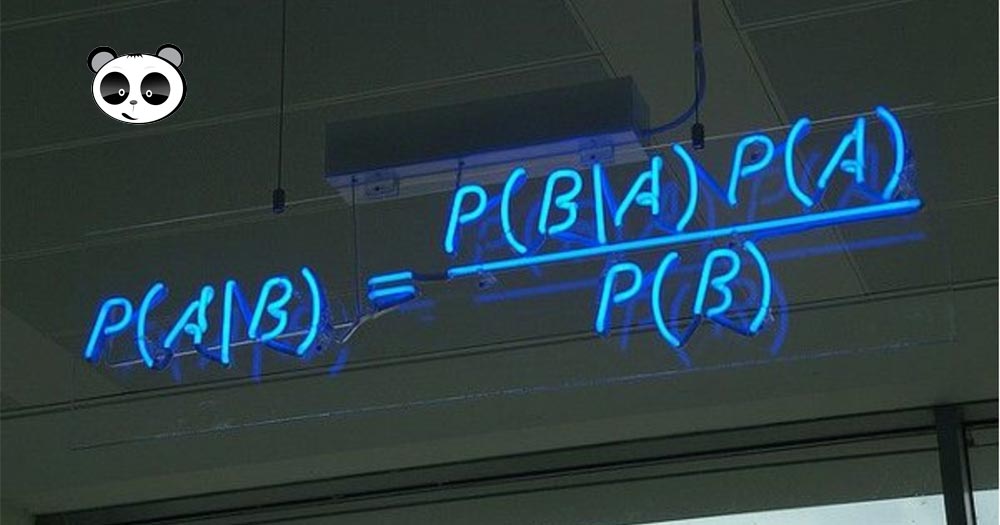


Hình 6: Thuật toán Học tăng cường

* **Phân loại tự động Bayes**

**Phân loại Bayes** dựa trên định lý Bayes với những giả định hoàn toàn độc lập (naïve) của các đặc tính. Từ đó cho ra các loại xác xuất đơn giản.

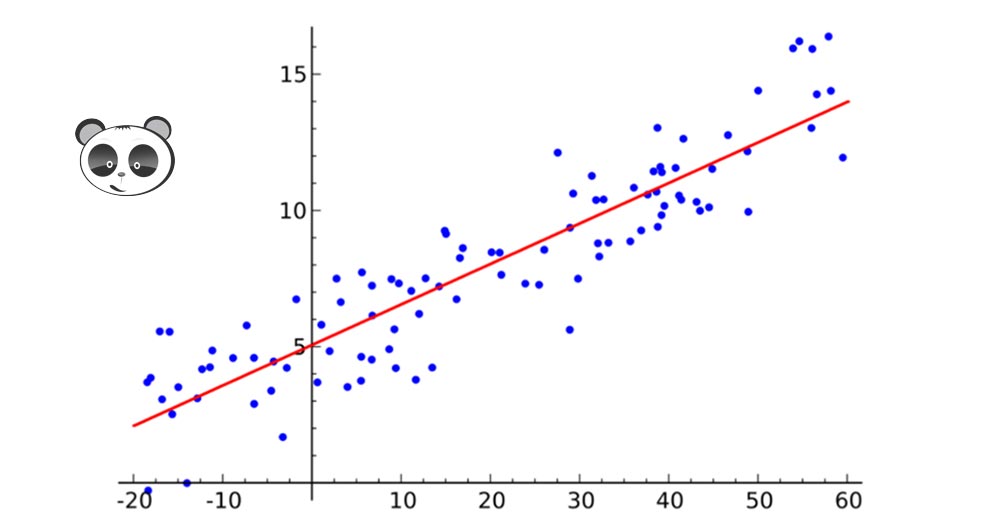
Thuật toán Machine Learning trên được ứng dụng trong một số bài toán sau:

* Đánh dấu email là spam hay không
* Phân loại bài viết tin tức thuộc các lĩnh vực như công nghệ, thể thao hay chính trị
* Kiểm tra đoạn văn để biết chúng mang tích cực hay tiêu cực
* Sử dụng cho những phần mềm nhận diện khuôn mặt…

Hình 7: Thuật toán phân loại tự động Bayes

* **Thuật toán Hồi quy tuyến tính ( Ordinary Least Squares Regression)**

Phương pháp được sử dụng để hồi quy trong trường hợp này chính là bình phương nhỏ nhất. Để đơn giản hóa bạn hãy nghĩ về hồi quy tuyến đơn giản chỉ là kẻ một đường thẳng đi qua tập hợp các điểm. Khi có được đường thẳng các bạn tiến hành đo khoảng cách thẳng đứng giữa điểm và đường thẳng. Đường mà các bạn lựa chọn chính là đường có độ dài nhỏ nhất. Ngoài phương pháp bình phương nhỏ nhất còn nhiều phương pháp mà các bạn có thể sử dụng.

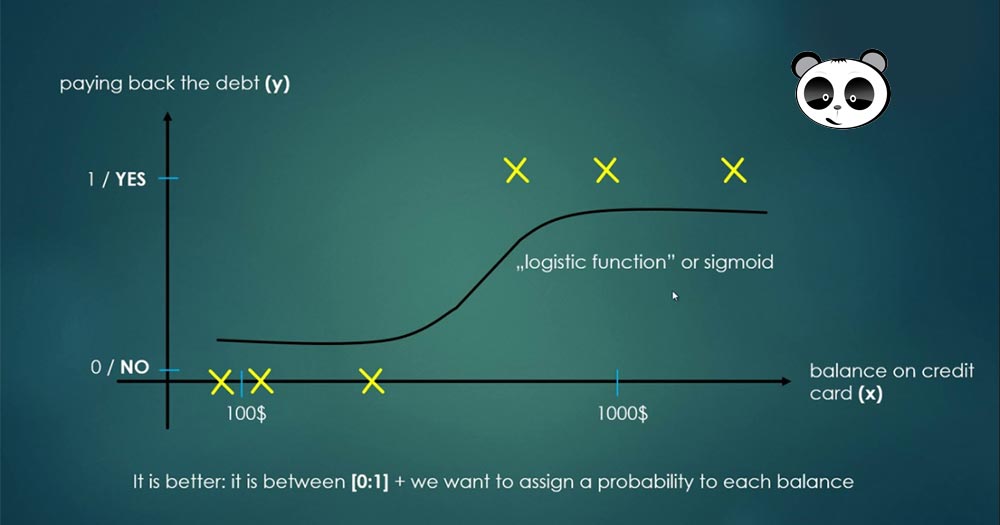
Mô hình này phù hợp cho các bài toán về dự đoán giá cả (chứng khoán, nhà đất,…)

Hình 8: Thuật toán quy hồi tuyến tính

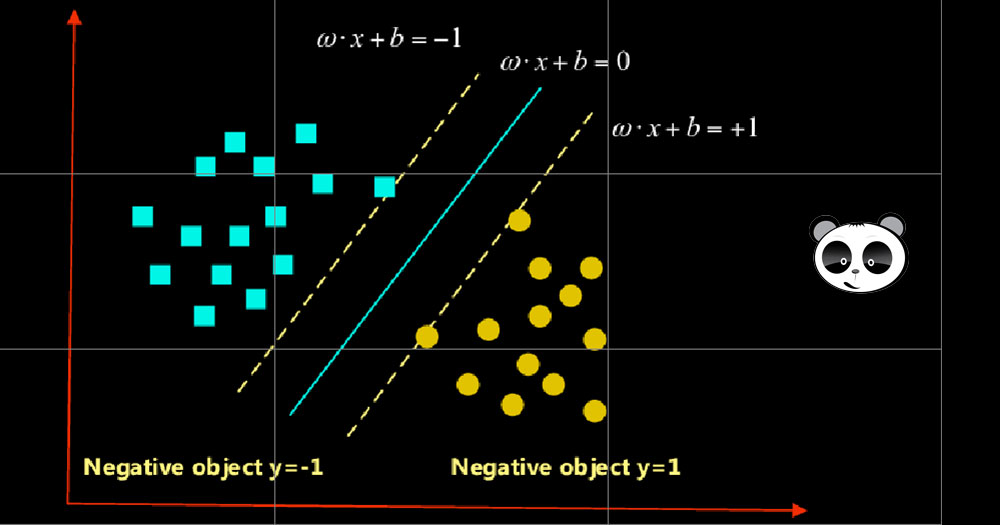
* **Thuật toán Hồi quy Logistic (Logistic Regression)** Hồi quy là một cách thống kế toàn bộ qua đó nhằm mô hình hóa cho một kết quả nhị thức với một hoặc nhiều hơn một biến giải thích. Cách làm này đo lường mối quan hệ giữa biến phụ thuộc phân loại hoặc nhiều biến độc lập với nhau thông qua cách ước tính xác suất của một hàm logistic, là sự phân bố tích lũy logistic.

Hồi quy được sử dụng trong các bài toán như sau:

* Đếm tín dụng (nhằm đưa ra quyết định nên hay không nên cho khách hàng vay vốn)
* Kiểm tra mức độ đạt được của một chiến dịch marketing
* Dự đoán doanh thu một sản phẩm cụ thể
* Dự đoán thời thời tiết: động đất,…

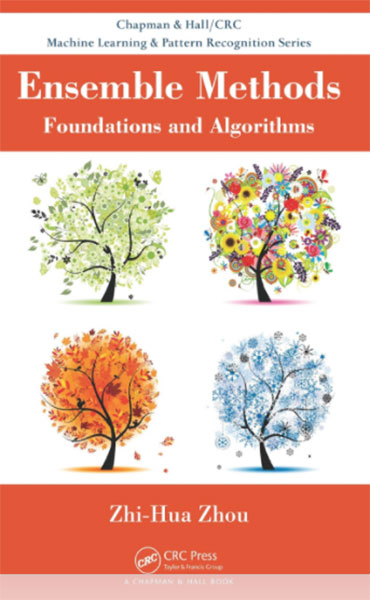


Hình 9: Thuật toán Quy hồi Logictics

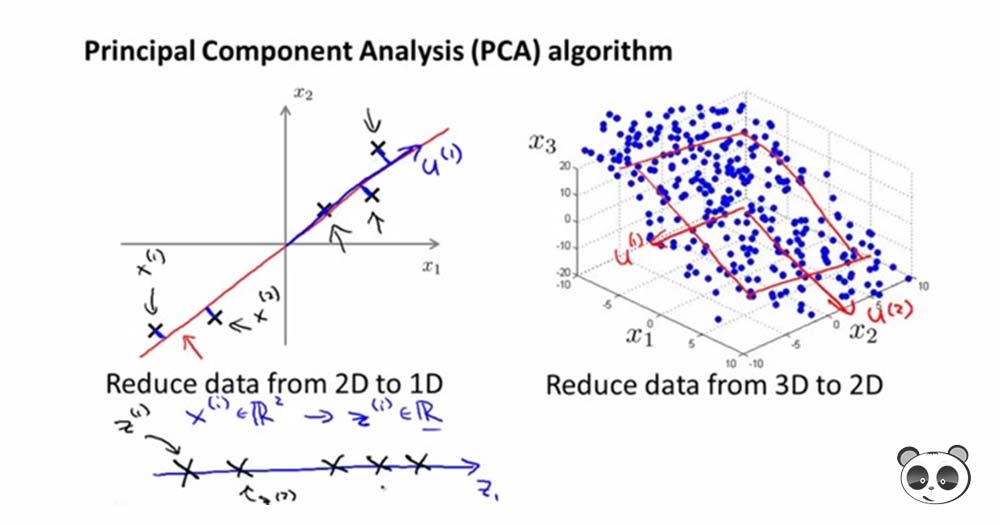
* **Support Vector Machines (SVM)** SVM được dùng cho bài toán nhị phân. Giả sử đầu bài cho tập hợp các điểm thuộc 2 loại trong môi trường N chiều, SVM là phương pháp để tìm ra N-1 mặt phẳng để ngăn các điểm thành 2 nhóm. Một ví dụ cụ thể đó là cho một tập hợp các thuộc 2 loại như hình mô tả phía dưới, SVM có nhiệm vụ tìm ra một đường thẳng để phân tách những điểm đó thành 2 loại sao cho độ dài khoảng cách giữa đường thẳng và những điểm là xa nhất có thể.

Hình 10: Thuật toán SVM

Xét trên quy mô SVM đã giải quyết được nhiều vấn đề lớn như hiển thị quảng cáo, phát hiện ra giới tính bằng hình ảnh, phân loại hình ảnh có phạm vi rộng (cần sửa đổi SVM cho phù hợp).

* **Kết hợp nhiều phương pháp (Ensemble Methods)** Đây là phương pháp tích hợp từ nhiều phương pháp khác nhau nhằm dự đoán kết quả. Qua đó rút ra kết luận dựa vào trọng số của từng phương pháp được áp dụng.
  + **Trung bình sai số (bias)**: với một số phương pháp các bạn sẽ nhận được sai số nhỏ, ngược lại có những phương pháp lại cho các bạn sai số lớn. Trung bình của hai hoặc nhiều phương pháp các bạn sẽ được một số chấp nhận được. Có thể là số được chấp nhận nhỏ hơn sai số khi sử dụng một phương pháp duy nhất.
  + **Giảm độ phụ thuộc vào tập dữ liệu (variance)**: tập hợp của nhiều ý kiến của nhiều mô hình sẽ tốt hơn là ý kiến của một mô hình. Đối với lĩnh vực tài chính đây là cách đa dạng hóa một – một danh mục hỗn hợp từ nhiều cổ phiếu sẽ thu được tính ổn định hơn là cổ phiếu riêng lẻ.
  + **Giảm over-fit**: là hiện tượng mà mô hình hoạt động kém với dữ liệu test nhưng lại tốt với training. Với việc sử dụng nhiều mô hình sẽ giảm tối đa được vấn đề nêu ra.

Hình 11: Thuật toán kết hợp nhiều phương pháp

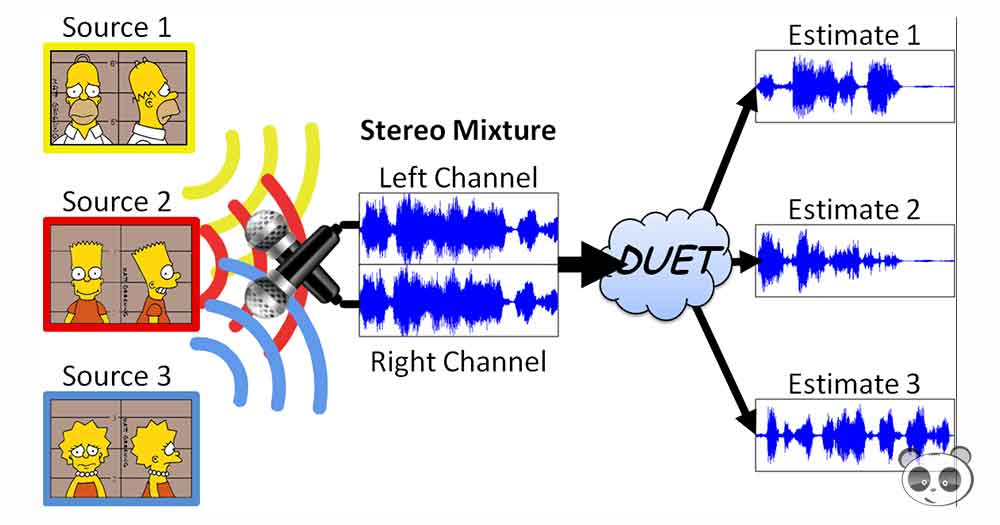
* **Phân tích dữ liệu thành phần cốt lõi (Principal Component Analysis-PCA) PCA là một thuật toán Machine Learning thống kê sử dụng phép biến đổi một tập hợp các dữ liệu từ một không gian nhiều chiều qua không gian ít chiều hơn. Việc làm này nhằm tối ưu việc thể hiện sự biến thiên của tập dữ liệu.**

Hình 12: Thuật toán phân tích dữ liệu phần kết lõi

Phép biến đổi này tạo ra những ưu điểm với dữ liệu như:

* Giảm tối đa số lượng không gian chứa dữ liệu nếu chúng có quá nhiều chiều khiến chúng ta khó hình dung.
* Loại bỏ trục tọa độ cũ thay vào đó là trục tọa độ mới và vẫn đảm bảo độ biến thiên của dữ liệu trên trục tọa độ mới đó.
* Tạo điều kiện để những liên kết bị ẩn có thể xuất hiện trên không gian mới.
* Đảm bảo những trục tọa độ trong không gian luôn trực quan đôi một, có thể trong không gian ban đầu các trục có thể khó nhìn.

PCA được ứng dụng trong một số bài toán bao gồm nén, đơn giản hóa luồng dữ liệu để thuận tiện cho quá trình học tập. Các bạn cần chú ý kiến thức miền rất quan trọng để các bạn đưa ra được quyết định có nên dùng PCA hay không. PCA không phù hợp trong trường hợp dữ liệu nhiễu.

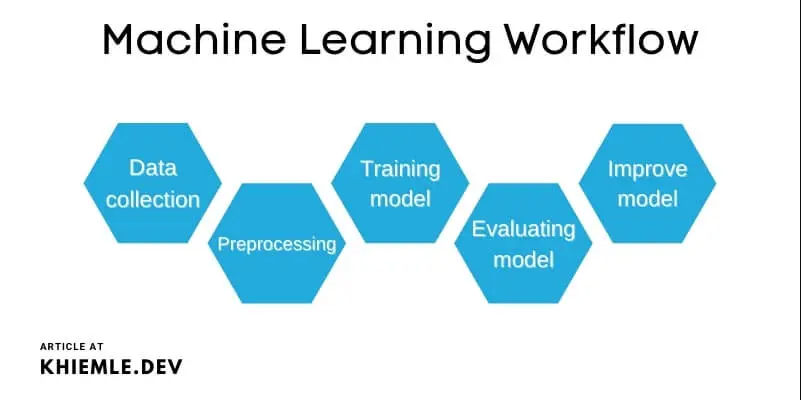
* **Phân tích dữ liệu thành phần độc lập (Independent Component Analysis)** ICA là kỹ thuật thống kê nhằm mục đích tìm ra các yếu tố còn ẩn dưới các bộ biến ngẫu nhiên, những phép đo hoặc tín hiệu. ICA định nghĩa một mô hình phát sinh cho dữ liệu đa biến quan sát được, và thường được đưa ra như một cơ sở dữ liệu lớn các mẫu. Trong mô hình những biến số liệu được giả định mang tính hỗn hợp tuyến tính của một biến tiềm ẩn bất kỳ, và hệ thống hỗn hợp cũng không rõ ràng. Những biến tiềm ẩn được gán gaussian và hoàn toàn độc lập với nhau, và chúng được gọi là các thành phần độc lập của tập dữ liệu được quan sát.

Hình 13: Thuật toán phân tích dữ liệu độc lập

ICA cũng có mối quan hệ với PCA, nhưng thực tế nó là một kỹ thuật mạnh hơn nhiều so với PCA. ICA có khả năng tìm ra những yếu tố tiềm ẩn của nguồn trong khi các phương pháp khác không đảm nhận được. ICA giúp hiển thị hình ảnh kỹ thuật số, cơ sở dữ liệu, những chỉ số liên quan đến kinh tế và đo lường tâm lý.

* + 1. **Machine Learning Workflow**

Machine learning workflow sẽ cho bạn thấy quy trình để làm việc với machine learning như thế nào. Hãy nhìn vào sơ đồ bên dưới:



Hình 14: Machine Learning WorkFlow

Cụ thể từng bước trong machine learning workflow như sau như sau:

1. **Data collection – thu thập dữ liệu:** để máy tính có thể học được bạn cần có một bộ dữ liệu (dataset), bạn có thể tự thu thập chúng hoặc lấy các bộ dữ liệu đã được công bố trước đó. Lưu ý là bạn phải thu thập từ nguồn chính thống, có như vậy dữ liệu mới chính xác và máy có thể học một cách đúng đắng và đạt hiệu quả cao hơn.
2. **Preprocessing – tiền xử lý:** bước này dùng để chuẩn hóa dữ liệu, loại bỏ các thuộc tính không cần thiết, gán nhãn dữ liệu, mã hóa một số đặc trưng, trích xuất đặc trưng, rút gọn dữ liệu nhưng vẫn đảm bảo kết quả… Bước này tốn thời gian nhất tỉ lệ thuận với số lượng dữ liệu bạn có. Bước 1 và 2 thường chiếm hơn 70% tổng thời gian thực hiện.
3. **Training model – huấn luyện mô hình:** bước này là bước bạn huấn luyện cho mô hình hay chính là cho nó học trên dữ liệu bạn đã thu thập và xử lý ở hai bước đầu.
4. **Evaluating model – đánh giá mô hình:** sau khi đã huấn luyện mô hình xong, chúng ta cần dùng các độ đo để đánh giá mô hình, tùy vào từng độ đo khác nhau mà mô hình cũng được đánh giá tốt hay không khác nhau. Độ chính xác của mô hình đạt trên 80% được cho là tốt.
5. **Improve – cải thiện:** sau khi đã đánh giá mô hình, các mô hình đạt độ chính xác không tốt thì cần được train lại, chúng ta sẽ lặp lại từ bước 3, cho đến khi đạt độ chính xác như kỳ vọng. Tổng thời gian của 3 bước cuối rơi vào khoảng 30% tổng thời gian thực hiện.
   * 1. **Ứng dụng của Machine Learning**

Machine learning được ứng dụng cực kỳ nhiều trong đời sống hiện nay trong mọi lĩnh vực:

* Tài chính – ngân hàng
* Sinh học
* Nông nghiệp
* Tìm kiếm, trích xuất thông tin
* Tự động hóa
* Robotics
* Hóa học
* Mạng máy tính
* Khoa học vũ trụ
* Quảng cáo
* Xử lý ngôn ngữ tự nhiên
* Thị giác máy tính

Và còn rất rất nhiều lĩnh vực mà machine learning có thể được áp dụng, machine learning tỏ ra cực kỳ hiệu quả, hơn hẳn con người trong cụ thể các lĩnh vực mà chúng được áp dụng.

* 1. **ML.Net**
     1. **Khái niệm .Net**
* Đây là một nền tảng phát triển đa nền tảng (Windows, Mac OS và Linux), mã nguồn mở và miễn phí, để xây dựng tất cả các loại ứng dụng: máy tính để bàn, thiết bị di động, web, trò chơi và cho Internet vạn vật.
* **.Net** cho phép sử dụng nhiều ngôn ngữ, trình soạn thảo và thư viện khác nhau cho các loại ứng dụng có thể có. Trong số các ngôn ngữ là:
* **C #:** Là một ngôn ngữ lập trình đơn giản, hiện đại, hướng đối tượng và an toàn.
* **F #:** Đây là một ngôn ngữ lập trình chức năng, đa nền tảng và mã nguồn mở cho .NET. Trong đó cũng bao gồm lập trình hướng đối tượng và lập trình mệnh lệnh.
* **Ngôn ngữ lập trình:** Đó là một ngôn ngữ có thể truy cập với cú pháp đơn giản để xây dựng các ứng dụng hướng đối tượng và an toàn kiểu.
* Đây là một nền tảng phát triển chéo tuyệt vời, vì nó cho phép các ứng dụng được tạo bằng các ngôn ngữ nói trên chạy nguyên bản trên bất kỳ Hệ điều hành tương thích nào, nhờ các triển khai tích hợp khác nhau, trong số đó có:
* **Lõi NET:** Triển khai đa nền tảng cho các trang web, máy chủ và ứng dụng bảng điều khiển là gì.
* **Nền tảng NET:** Tương thích với các trang web, dịch vụ, ứng dụng máy tính để bàn và nhiều hơn nữa trên Windows.
* **Xamarin / Mono:** Triển khai .NET là gì để chạy các ứng dụng trên Hệ điều hành di động chính.
* **.Net** chứa một bộ API cơ bản chung cho tất cả các triển khai .NET. Ngoài ra, mỗi triển khai cũng có thể hiển thị các API bổ sung dành riêng cho Hệ điều hành mà chúng sẽ chạy. Ví dụ, .NET Framework là một triển khai .NET chỉ dành cho Windows bao gồm các API để truy cập Windows Registry.
* **.Net** có một Thư viện khổng lồ (hệ sinh thái các gói) mở rộng chức năng của nó. Để sử dụng chúng, bạn có thể sử dụng NuGet, là một trình quản lý gói được xây dựng đặc biệt cho **«.NET»** chứa hơn 90.000 gói.
  + 1. **Khái niệm ML.Net**

**ML.NET** là một thư viện học máy phần mềm miễn phí cho các ngôn ngữ lập trình C # và F #. Nó cũng hỗ trợ các mô hình Python khi được sử dụng cùng với NimbusML. Các tác vụ ML bổ sung như hệ thống phát hiện và đề xuất bất thường đã được thêm vào và các phương pháp tiếp cận khác như học sâu sẽ được đưa vào các phiên bản trong tương lai.

Hình 15: ML.Net

ML.NET mang lại khả năng phân tích và dự đoán Machine Learning dựa trên mô hình cho các nhà phát triển .NET hiện có. Khung được xây dựng dựa trên .NET Core và .NET Standard kế thừa khả năng chạy đa nền tảng trên Linux, Windows và macOS. Mặc dù khung ML.NET là mới, nguồn gốc của nó bắt đầu vào năm 2002 như là một dự án nghiên cứu của Microsoft có tên TMSN (tìm kiếm và điều hướng khai thác văn bản) để sử dụng nội bộ trong các sản phẩm của Microsoft. Sau đó nó được đổi tên thành TLC (mã học tập) vào khoảng năm 2011. ML.NET có nguồn gốc từ thư viện TLC và phần lớn đã vượt qua cha mẹ của nó cho biết Tiến sĩ James McCaffrey, Nghiên cứu của Microsoft cho biết.

**ML.NET** bao gồm các thành phần như Model Builder (công cụ Giao diện người dùng Visual Studio), CLI (giao diện dòng lệnh) và AutoML (Automated Machine Learning được sử dụng để xây dựng các mô hình tùy chỉnh).

* + 1. **Các công cụ hổ trợ trong ML.Net**
* **TimeSeries hỗ trợ dự báo và phát hiện bất thường:**

Các nhà phát triển có thể sử dụng gói Microsoft.ML.TimeSeries cho nhiều tình huống như: phát hiện đột biến và thay đổi doanh số bán sản phẩm bằng cách sử dụng mô hình phát hiện bất thường hoặc tạo dự báo doanh số có thể bị ảnh hưởng bởi tính thời vụ và bối cảnh liên quan đến thời gian khác

* **Các gói ML.NET để sử dụng các mô hình TensorFlow và ONNX:**

ML.NET đã được thiết kế như một nền tảng mở rộng để bạn có thể tiêu thụ các mô hình ML phổ biến khác như mô hình TensorFlow và ONNX và có quyền truy cập vào nhiều kịch bản học máy và học sâu hơn, như phân loại hình ảnh, phát hiện đối tượng và hơn thế nữa.

* **Các mô hình ML.NET trong các ứng dụng web hoặc không có máy chủ với gói tích hợp Microsoft.Extensions.ML:**

Gói này giúp dễ dàng tích hợp mô hình tải ML.NET để ghi điểm trong các ứng dụng ASP.NET, Azure Functions và các dịch vụ web. Cụ thể, gói này cho phép nhà phát triển sử dụng Microsoft.Extensions.ML để tải mô hình ML.NET bằng Dependency Injection và tối ưu hóa hiệu suất thực thi và hiệu suất của mô hình trong môi trường đa luồng như ASP.NET ứng dụng Core.

* **ML.NET CLI:**

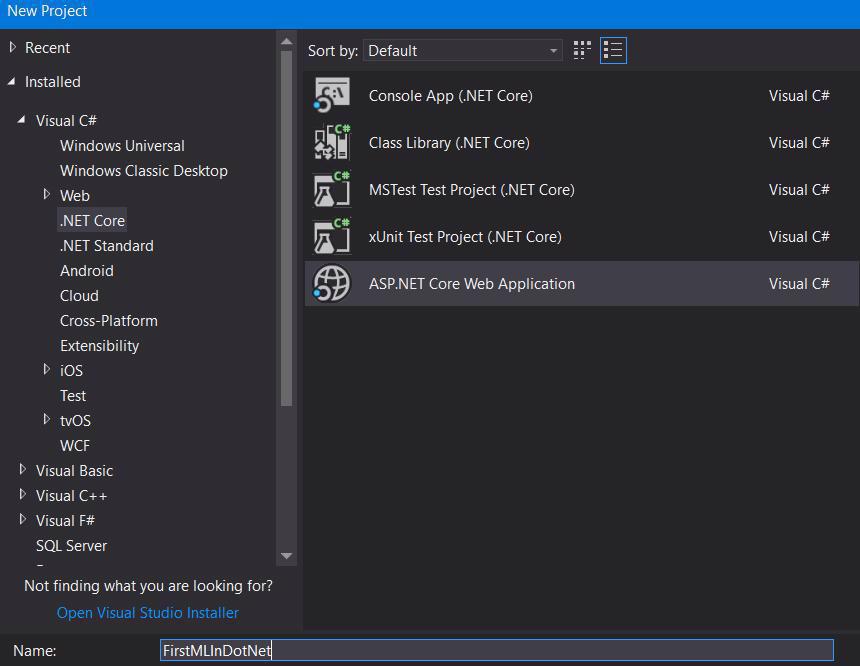
Bạn có thể sử dụng ML.NET CLI để tự động tạo mô hình ML.NET và mã C # cơ bản. Bạn có thể chạy ML.NET CLI trên bất kỳ lời nhắc lệnh nào (Windows, Mac hoặc Linux).

* + 1. **Các ứng dụng của ML.Net**

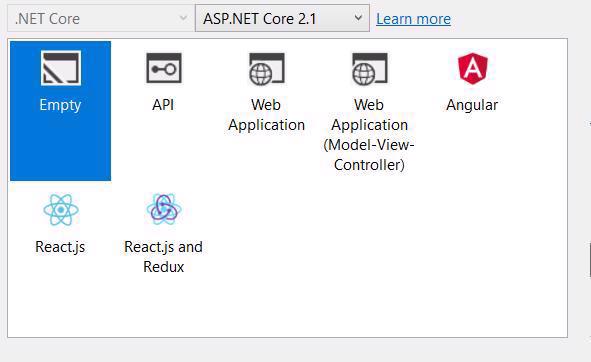
**ML.Net** cũng tương tự như các ứng dụng của Machine Learning:

* Phân tích tình cảm
* Đề xuất sản phẩm
* Dự đoán giá
* Phân khúc khách hàng
* Phát hiện đối tượng
* Phát hiện gian lận
* Phát hiện doanh số tăng đột biến
* Phân loại hình ảnh
* Dự báo doanh số
  + 1. **Cách cài đặt thư viện ML.Net**
* Cài đặt gói ML.NET NuGet

Trước tiên, hãy đảm bảo rằng bạn đã cài đặt Visual Studio bản mới nhất và cài đặt .NET Core 2.0 trở lên. ML.NET cũng hoạt động trên .NET Framework 4.7 trở lên. Lưu ý rằng ML.NET hiện phải chạy trong quy trình 64 bit.

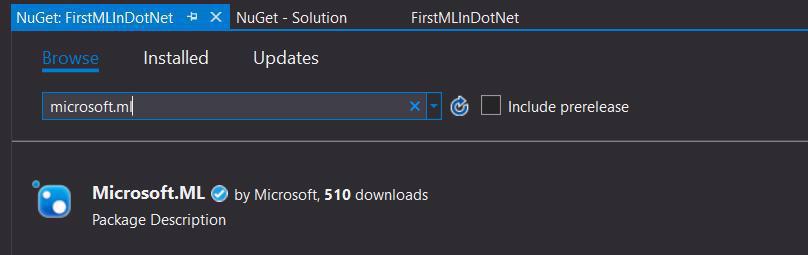
Khi bạn đã cài đặt tất cả những thứ này, hãy mở Visual Studio -> Tạo dự án mới -> Chọn ứng dụng Core Web:

Hình 16: Tạo Project

Chọn dự án Empty:

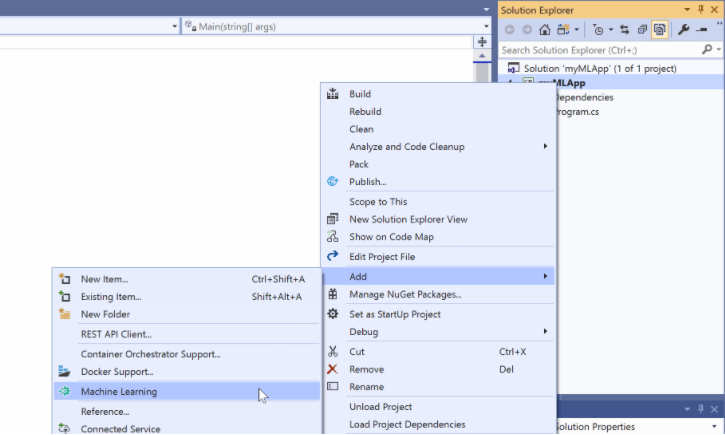
Hình 17: Chọn dự án

Khi ứng dụng được tạo, chúng ta hãy thêm gói NuGet Microsoft.ML cần thiết.

Tìm kiếm với Microsoft Microsoft.ML, trong Trình quản lý gói NuGet và nhấp vào Cài đặt:

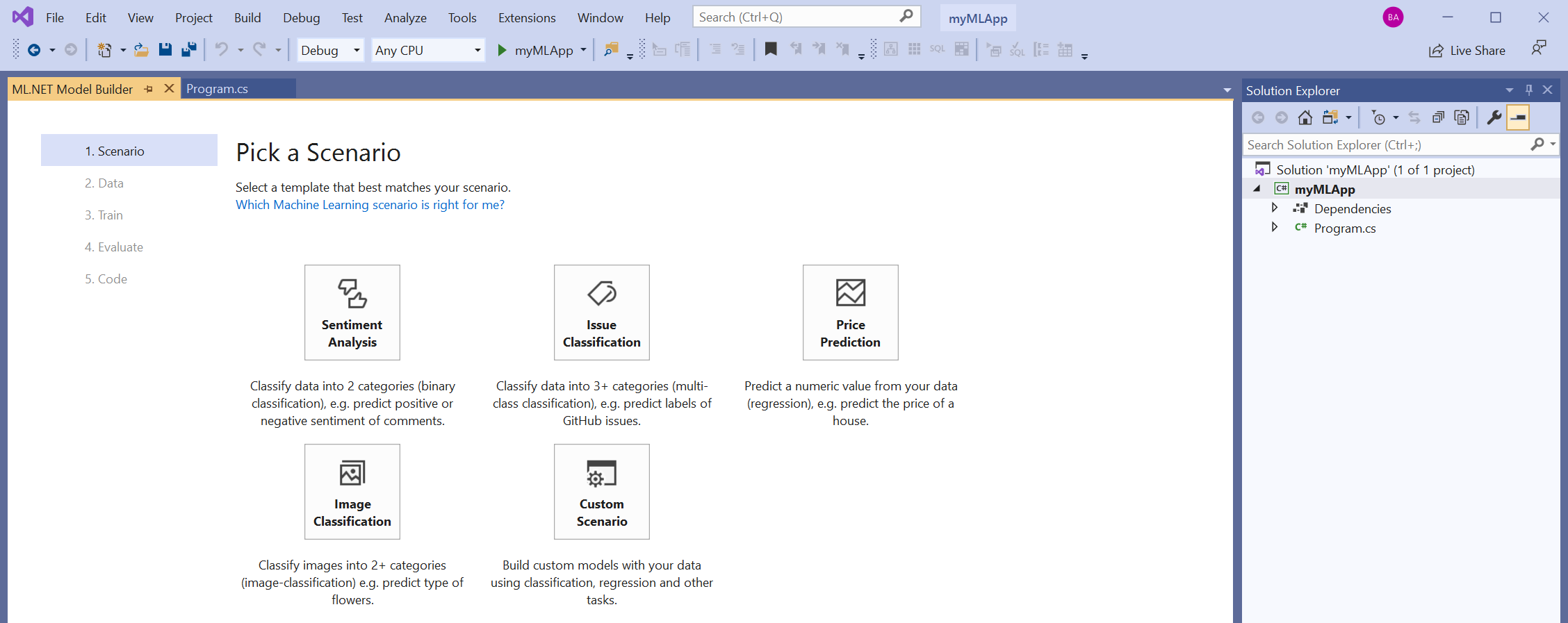
Hình 18: Cài đặt gói NuGet

* **Bắt đầu xây dựng chương trình**

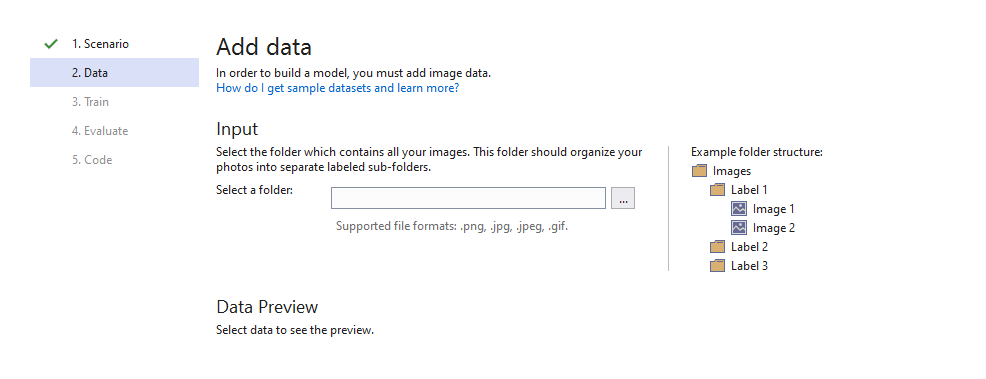
Để bắt đầu sử dụng ML.NET Model Builder, các bạn làm như sau: **Click chuột phả**i lên **Project** của bạn (hay tên tuỳ ý của bạn đã tạo ở bước trên) trong **Solution Explorer**và chọn **Add** > **Machine Learning**.

Hình 19: Add Machine Learning

Cửa sổ **ML.NET Model Builder** sẽ được mở ra trong VS. Như hình dưới đây:

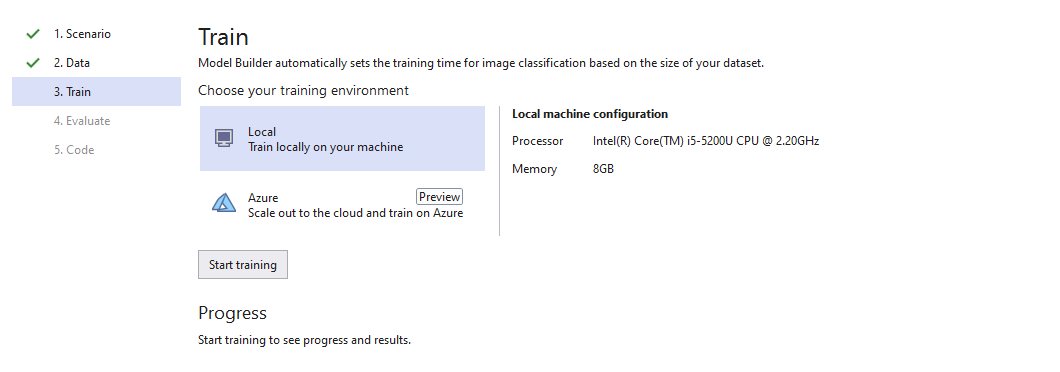


Hình 20: Chọn loại ứng dụng

Tại đây, có rất nhiều ứng dụng của ML.NET bạn chọn ứng dựng mà bạn muốn sử dụng.

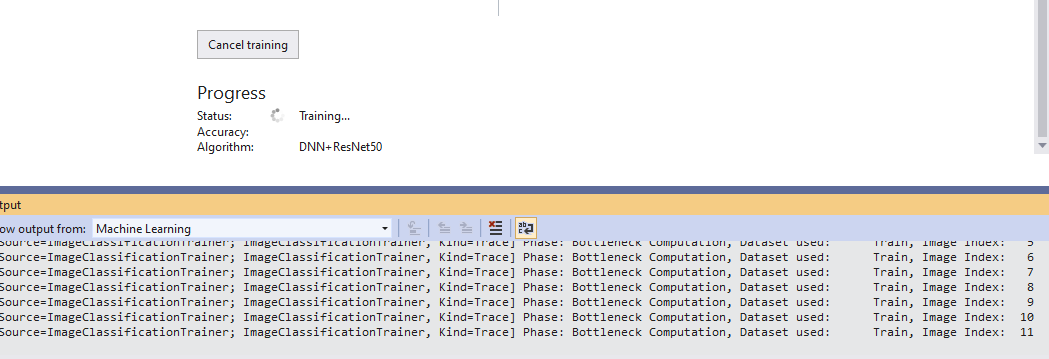
Hình 21: Add File cho Máy học

Chương trình sẽ yêu cầu bạn nhập mục chứa dữ liệu để vào máy để Trainning.

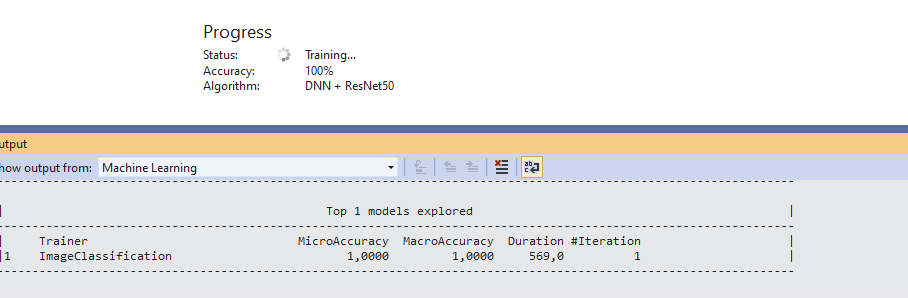
Tiếp đến, các bạn tiến hành bấm Train để dạy cho máy.

Hình 22: Chọn loại nền tảng để Máy học

Tại đây, các bạn có 2 tuỳ chọn trên **Azure** hoặc **Local, bạn chọn 1 trong 2.**

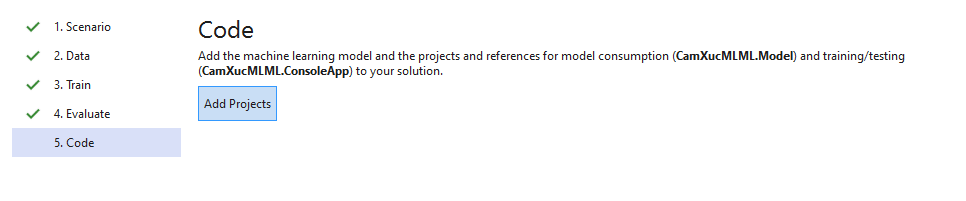
Sau đó, các bạn bấm **Start trainning** và máy bắt đầu học, các bạn ngồi đợi nhé dữ nhiều càng nhiều thì thời gian máy học càng lâu, trong lúc máy học sẽ **chiếm nhiều CPU và RAM**.

Hình 23: Máy tiến hành Training

Sau khi máy học thành công sẽ trả về kết quả sau:

Hình 24: Kết quả Training

* Tiến hành code

Tiếp đến, các bạn kéo xuống và bấm vào **Code**để thêm vào Project của bạn.

Hình 25: Tiến hành Code

Các bạn ấn **Add Projects** là xong bước xây dựng dữ liệu nha!

Lúc các bạn thêm **Add Projects,**nó sẽ tạo ra thêm một <tên project của bạn>.ConsoleApp.

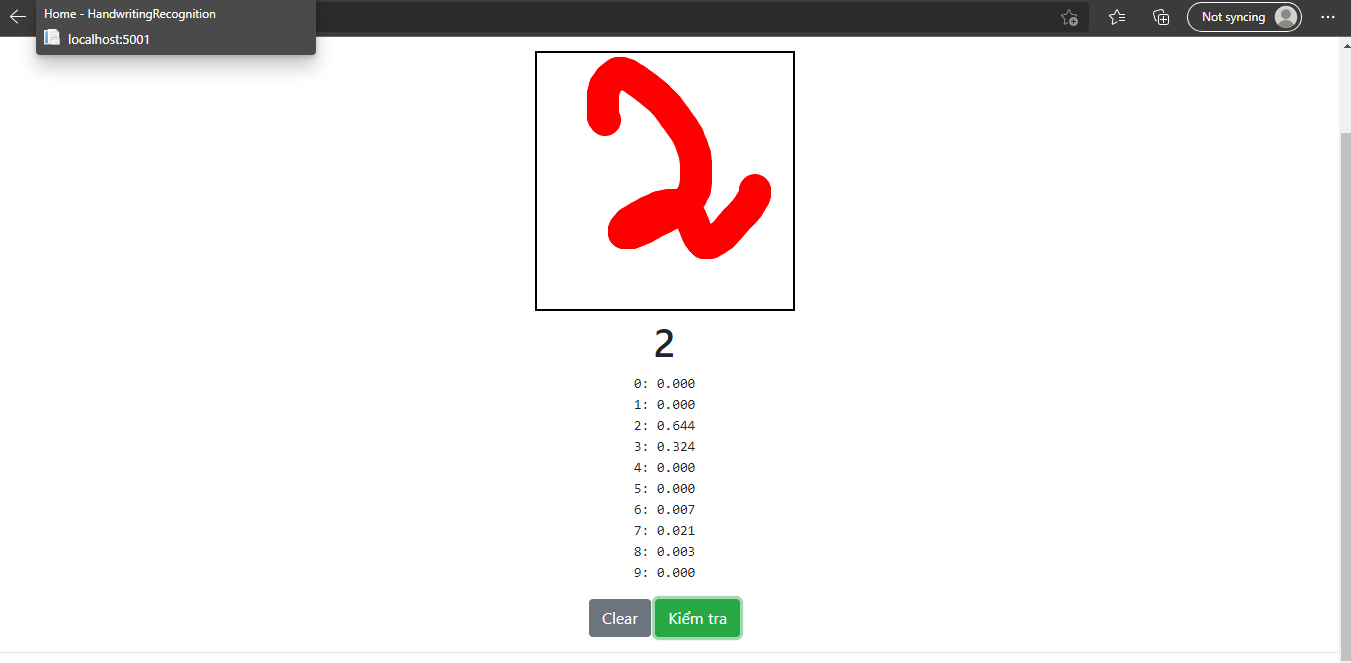
Sau đó tiến hành code chương trình.

**CHƯƠNG 2: CHƯƠNG TRÌNH” NHẬN DIỆN CHỮ VIẾT TAY”**

Từ những cơ sở lý thuyết, phương pháp phân tích được đề cập ở chương 1. Em đã xây dựng và phát triển thành công ứng dụng đơn giản trên môi trường Visual studio 2019, viết bằng ngôn ngữ lập trình C#. Sử dụng công cụ và thư viện mở dành cho C# là ML.Net.

1. **Chương trình “ Nhận chữ viết tay”**

* Vẽ vào ô Sau đó bấm kiểm tra để chương trình dự đoán.



Hình 26: Nhận diện chữ viết tay

# **CHƯƠNG 3: KẾT LUẬN**

* + - 1. **Kết quả đạt được**
* Về tri thức:

Trong suốt khoảng thời gian 2 tuần, em đã học được rất nhiều về kiến thức nền tảng, các phương pháp tìm và đọc tại liệu. Các ứng dụng và khả năng của công cụ Microsoft ML.Net.

* Về ứng dụng:

Đã đạt được các yêu cầu đề về nhận diện chữ viết tay trong ML.Net như:

* Chương trình **Nhận diện, phát hiện và hiển thị chữ số**.
* Về con người:

Khoảng thời gian 2 tuần là khoảng thời gian đủ để em hoàn thành tiểu luận, mặc dù cũng gặp đôi chút khó khăn. Tuy nhiên em đã hoàn thành xong bài tiểu luận được giao.

* + - 1. **Hạn chế tồn tại**

Vì thời gian thực hiện đề tài cũng như kiến thức của bản thân là có giới hạn nên vẫn còn vài thiếu sót như: Giao diện chưa đẹp mắt, các phương pháp chưa phong phú, khả năng xử lí chưa thực sự chính xác, Chương trình còn nhiều lỗi cần khắc phục và cải thiện thêm tính năng mới.

Rất mong thầy,cô sẽ bỏ qua sự thiếu sót của nhóm em. Em xin chân thành cảm ơn !

# **TÀI LIỆU THAM KHẢO**

* + - 1. **ML.Net:** [ML.NET – Wikiwand](https://www.wikiwand.com/en/ML.NET).
      2. **^** Ankit Asthana (2017-05-07). [*"Giới thiệu ML.NET: Khung học máy đa nền tảng, đã được chứng minh và nguồn mở"*](https://blogs.msdn.microsoft.com/dotnet/2018/05/07/introducing-ml-net-cross-platform-proven-and-open-source-machine-learning-framework/). blogs.msdn.microsoft.com..
      3. [*"ML.NET: Machine Learning được thực hiện cho .NET"*](https://www.microsoft.com/net/learn/apps/machine-learning-and-ai/ml-dotnet). Microsoft..
      4. David Ramel (2018-05-08). [*"Nguồn mở, đa nền tảng ML.NET đơn giản hóa học máy - Tạp chí Visual Studio"*](https://visualstudiomagazine.com/articles/2018/05/08/ml-net-framework.aspx). Tạp chí Visual Studio.