Lista kontrolna projektu uczenia maszynowego

Poniższa lista stanowi pomoc podczas tworzenia projektów uczenia maszynowego. Na proces ten składa się osiem etapów:

1. Określenie problemu i przeanalizowanie go w szerszej perspektywie.
2. Pozyskanie danych.
3. Analiza danych w celu wykrycia dodatkowych informacji.
4. Przygotowanie danych w sposób uwidaczniający wzorce wykorzystywane przez algorytmy uczenia maszynowego.
5. Sprawdzenie wielu modeli i stworzenie krótkiej listy najwydajniejszych z nich.
6. Dostrojenie modeli i połączenie ich w zespoły uzyskujące jeszcze lepsze wyniki.
7. Zaprezentowanie rozwiązania.
8. Uruchomienie, monitorowanie i utrzymywanie systemu.

Oczywiście, każdą listę należy dowolnie dostosowywać do potrzeb danego projektu.

Określenie problemu i przeanalizowanie go w szerszej perspektywie

1. Zdefiniuj cel w kategoriach biznesowych.
2. W jaki sposób będzie używane Twoje rozwiązanie?
3. Jakie istnieją obecnie rozwiązania/obejścia (jeśli istnieją)?
4. W jakich kategoriach należy zdefiniować problem (nienadzorowany/nadzorowany, przyrostowy/statyczny itd.)?
5. W jaki sposób będzie mierzona wydajność modelu?
6. Czy pomiar wydajności jest powiązany z celem biznesowym?
7. Jaka jest minimalna wydajność wymagana do spełnienia celu biznesowego?

I S B N978-83-283-4373-3

8. Czy istnieją jakieś porównywalne problemy? Czy możesz wykorzystać dostępne doświadczenia lub narzędzia?
9. Czy możesz skorzystać z pomocy ekspertów?
10. W jaki sposób można ręcznie rozwiązać dany problem?
11. Sporządź listę założeń ustalonych przez Ciebie (lub innych).
12. W miarę możliwości zweryfikuj założenia.

Pozyskanie danych

Uwaga: W miarę możliwości zautomatyzuj ten etap, aby móc łatwo uzyskiwać świeże dane.

1. Określ rodzaj i ilość potrzebnych danych.
2. Wyznacz miejsce, w którym możesz uzyskać dane, i udokumentuj je.
3. Sprawdź, jak wiele przestrzeni dyskowej będzie potrzebne na przechowywanie danych.
4. Sprawdź zobowiązania prawne i w razie potrzeby uzyskaj autoryzację.
5. Zdobądź uprawnienia dostępu.
6. Stwórz przestrzeń roboczą (z wystarczającą pojemnością dyskową).
7. Pozyskaj dane.
8. Przekształć dane do formatu, który umożliwi łatwą manipulację nimi (bez zmieniania istoty samych danych).
9. Upewnij się, że wrażliwe dane zostały usunięte lub zabezpieczone (np. zamaskowane).
10. Sprawdź rozmiar i typ danych (szeregi czasowe, próbki, dane geograficzne itd.).
11. Wydziel zestaw testowy, odłóż go i nigdy do niego nie zaglądaj (żadnego podglądania danych!).

Analiza danych

Uwaga: Spróbuj uzyskać na tym etapie wsparcie eksperta z danej dziedziny.

1. Stwórz kopię analizowanych danych (w razie potrzeby przepróbkowując je do rozsądnych rozmiarów).
2. Stwórz notatnik Jupyter, w którym będziesz przechowywać wyniki analizy danych.
3. Określ każdy atrybut i jego parametry:
   a. Nazwę.
   b. Typ (kategorialne, stało-/zmiennoprzecinkowe, ograniczone/nieograniczone, tekstowe, strukturalne itd.).
   c. Odsetek brakujących wartości.
   d. Zaszumienie i rodzaj szumu (stochastyczny, elementy odstające, błędy zaokrąglenia itd.).
   e. Przydatność w określonym zadaniu.
   f. Rodzaj rozkładu (gaussowski, jednorodny, logarytmiczny itd.).

I S B N978-8328343733

4. W przypadku zadań uczenia nadzorowanego określ docelowy atrybut (docelowe atrybuty).
5. Zwizualizuj dane.
6. Przeanalizuj korelacje pomiędzy atrybutami.
7. Zastanów się, w jaki sposób można ręcznie rozwiązać problem.
8. Określ obiecujące przekształcenia, które mogą zostać zastosowane.
9. Określ dodatkowe dane, które mogą okazać się przydatne (patrz etap „Pozyskanie danych").
10. Udokumentuj zgromadzoną wiedzę.

Przygotowanie danych

Uwagi:

- Pracuj na kopiach danych (oryginalny zbiór danych powinien zostać nietknięty).
- Napisz funkcje dla wszystkich przeprowadzanych przekształceń; wynika to z pięciu powodów:
  - Aby można było łatwiej przygotowywać świeże dane.
  - Aby można było wprowadzać te przekształcenia w przyszłych projektach.
  - Aby oczyścić i przygotować zestaw testowy.
  - Aby oczyszczać i przygotowywać nowe próbki po wdrożeniu projektu do środowiska produkcyjnego.
  - Aby można było traktować te funkcje przekształceń jako hiperparametry.

1. Oczyszczanie danych:
   a. Dostosuj lub usuń elementy odstające (nieobowiązkowe).
   b. Uzupełnij brakujące wartości (np. zerami, średnią, medianą…) lub usuń odpowiednie rzędy (albo kolumny).
2. Dobór cech (nieobowiązkowy):
   a. Usuń atrybuty, które nie dostarczają użytecznych informacji do wykonania zadania.
3. W miarę możliwości inżynieria cech:
   a. Zdyskretyzuj cechy ciągłe.
   b. Dokonaj rozkładu cech (np. kategorialne, data/godzina itd.).
   c. Dodaj obiecujące przekształcenia cech (np. $\log(x)$, $\text{sqrt}(x)$, x^2 itd.).
   d. Połącz cechy w nowe, obiecujące cechy.
4. Skalowanie cech: standaryzuj lub normalizuj cechy.

I S B N978-83-283-4373-3

Stworzenie krótkiej listy obiecujących modeli

Uwagi:

- Jeżeli zbiór danych jest bardzo duży, możesz chcieć próbkować mniejsze zestawy uczące, dzięki czemu możesz trenować wiele różnych modeli w rozsądnie krótkim czasie (pamiętaj, że to rozwiązanie nie jest dobre dla złożonych modeli, takich jak duże sieci neuronowe lub losowe lasy).
- Postaraj się zautomatyzować również ten etap.

1. Wyucz wiele różnych testowych wersji modeli (np. liniowe, naiwne bayesowskie, maszyny SVM, losowy las, sieć neuronowa itd.) za pomocą standardowych parametrów.
2. Zmierz i porównaj wydajność tych modeli.
   - Dla każdego modelu wykonaj N-krotny sprawdzian krzyżowy oraz oblicz średnią i odchylenie standardowe miary wydajności dla N podzbiorów.
3. Przeanalizuj najistotniejsze zmienne każdego algorytmu.
4. Przeanalizuj rodzaje błędów popełnianych przez modele.
   - Jakie dane wykorzystałby człowiek do uniknięcia tych błędów?
5. Wykonaj szybki przebieg doboru i inżynierii cech.
6. Wykonaj jeszcze jeden albo dwa dodatkowe przebiegi pięciu powyższych czynności.
7. Sporządź krótką listę od trzech do pięciu najbardziej obiecujących modeli, najlepiej takich, które popełniają różne rodzaje błędów.

Dostrojenie modelu

Uwagi:

- Na tym etapie należy wykorzystać jak największą ilość danych, zwłaszcza pod koniec strojenia.
- Jak zwykle postaraj się zautomatyzować jak największą część tego procesu.

1. Dostrój hiperparametry za pomocą sprawdzianu krzyżowego.
   a. Potraktuj dobrane funkcje przekształceń danych jako hiperparametry modelu, zwłaszcza jeśli nie masz co do nich pewności (np. powinnam/powinienem zastąpić brakujące wartości zerami czy medianą? A może po prostu usunąć rzędy?).
   b. Zawsze wybieraj losowe przeszukiwanie zamiast przeszukiwania siatki, jeśli dostępnych jest niewiele wartości hiperparametrów. Jeżeli proces uczenia trwa bardzo długo, lepszym rozwiązaniem może okazać się optymalizacja bayesowska (np. za pomocą procesów gaussowskich, co zostało opisane przez Jaspera Snoeka, Hugo Larochelle'a i Ryana Adamsa, *https://arxiv.org/pdf/1206.2944.pdf*)[1].
2. Wypróbuj metody zespołowe. Zbiór połączonych najlepszych modeli często osiąga lepsze rezultaty od jego poszczególnych składowych.

[1] *Practical Bayesian Optimization of Machine Learning Algorithms*, J. Snoek, H. Larochelle, R. Adams (2012).

I S B N978-8328343733

3. Gdy już będziesz zadowolona/zadowolony ze swojego modelu, zmierz jego wydajność za pomocą zestawu testowego, aby określić błąd generalizacji.

Nie dostrajaj modelu po zmierzeniu jego błędu uogólniania: spowodowałoby to przetrenowanie wobec zbioru testowego.

Zaprezentowanie rozwiązania

1. Udokumentuj postępy i dokonania.
2. Stwórz elegancką prezentację.
   - Najpierw zaprezentuj problem w szerszej perspektywie.
3. Wyjaśnij, dlaczego Twoje rozwiązanie spełnia cel biznesowy.
4. Nie zapomnij zaprezentować ciekawych spostrzeżeń dokonanych w trakcie pracy nad projektem.
   a. Opisz rozwiązania, które zadziałały i które okazały się nieskuteczne.
   b. Wymień ustanowione założenia i ograniczenia systemu.
5. Upewnij się, że najważniejsze odkrycia zostaną przekazane za pomocą ślicznych wizualizacji lub przystępnych stwierdzeń (np. „mediana dochodów stanowi główny predyktor cen domów”).

Do dzieła!

1. Przygotuj rozwiązanie pod środowisko produkcyjne (podłącz pod wejścia danych produkcyjnych, napisz jednostki testujące itd.).
2. Napisz kod monitorowania sprawdzający wydajność systemu w regularnych odstępach czasu i wysyłający alerty, gdy ta spadnie.
   a. Pamiętaj o zjawisku powolnej degradacji: modele ulegają „rozkładowi” wraz z ewoluowaniem danych.
   b. Pomiar wydajności może wymagać czynnika ludzkiego na którymś etapie potoku (np. poprzez usługi źródeł społecznościowych).
   c. Monitoruj również jakość danych wejściowych (np. niesprawny czujnik wysyłający losowe wartości lub brak dynamiki danych dostarczanych przez zespół znajdujący się na wcześniejszym etapie potoku). Jest to szczególnie istotne w przypadku systemów uczenia przyrostowego.
3. Trenuj regularnie modele za pomocą świeżych danych (w miarę możliwości zautomatyzuj ten proces).

I S B N978-8328343733