Out[3]: ASIMOV

O checklist de um projeto de Machine Learning

Este é o checklist de verificação para orientá-lo em seus projetos de Machine Learning. Existem oito etapas principais:

1. **Reflita sobre o problema e observe o quadro geral.**
2. **Obtenha os dados.**
3. **Explore os dados para obter insights.**
4. **Prepare os dados para expor melhor os padrões de dados subjacentes aos algoritmos de aprendizado de máquina.**
5. **Explore muitos modelos diferentes e selecione os melhores.**
6. **Ajuste seus modelos e combine-os em uma ótima solução.**
7. **Apresente sua solução.**
8. **Inicie, monitore e mantenha seu sistema.**

Obviamente, você deve se sentir à vontade para adaptar esta lista de verificação às suas necessidades.

Abaixo, vamos estratificar cada etapa.

1. Reflita sobre o problema e observe o quadro geral.

1. Defina o objetivo do ponto de vista do negócio.
2. Como sua solução será usada?
3. Quais são as soluções / soluções alternativas atuais (se houver)?
4. Como você deve enquadrar este problema (supervisionado / não supervisionado, online / offline, etc.)?
5. Como o desempenho deve ser medido? ”
6. A medida de desempenho está alinhada com o objetivo do negócio?
7. Qual seria o desempenho mínimo necessário para atingir o objetivo do negócio?
8. O que são problemas comparáveis? Você pode reutilizar experiências ou ferramentas?
9. A experiência humana está disponível?
10. Como você resolveria o problema manualmente?
11. Liste as suposições que você (ou outras pessoas) fizeram até agora.
12. Verifique as suposições, se possível.

2. Obtenha os dados.

Observação: automatize o máximo possível para obter dados atualizados com facilidade.

1. Liste os dados de que você precisa e de quanto você precisa.
2. Encontre e documente onde você pode obter esses dados.
3. Verifique quanto espaço vai ocupar.
4. Verifique as obrigações legais e obtenha autorização, se necessário.
5. Obtenha autorizações de acesso.
6. Crie um espaço de trabalho (com espaço de armazenamento suficiente).
7. Obtenha os dados.

8. Converta os dados em um formato que você possa manipular facilmente (sem alterar os próprios dados).
9. Certifique-se de que as informações confidenciais sejam excluídas ou protegidas (por exemplo, tornadas anônimas).
10. Verifique o tamanho e o tipo de dados (série temporal, amostra, geográfico, etc.).
11. Experimente um conjunto de teste, coloque-o de lado e nunca olhe para ele (sem espionagem de dados!).

3. Explore os dados para obter insights.

Observação: tente obter informações de um especialista de campo para essas etapas.

1. Crie uma cópia dos dados para exploração (amostragem até um tamanho gerenciável, se necessário).
2. Crie um Notebook Jupyter para manter um registro de sua exploração de dados.
3. Estude cada atributo e suas características:
    - Nome
    - Tipo (categórico, int / float, limitado / ilimitado, texto, estruturado, etc.)
    - % de valores ausentes
    - Ruído e tipo de ruído (estocástico, outliers, erros de arredondamento, etc.)
    - Utilidade para a tarefa em questão
    - Tipo de distribuição (gaussiana, uniforme, logarítmica, etc.)
4. Para tarefas de aprendizagem supervisionada, identifique o (s) atributo (s) alvo.
5. Visualize os dados.
6. Estude as correlações entre atributos.
7. Estude como você resolveria o problema manualmente.
8. Identifique as transformações promissoras que você pode querer aplicar.
9. Identifique dados extras que seriam úteis (volte para “Obter os Dados”).
10. Documente o que você aprendeu.

4. Prepare os dados para expor melhor os padrões de dados subjacentes aos algoritmos de aprendizado de máquina.

Notas:

- Trabalhe em cópias dos dados (mantenha o conjunto de dados original intacto).
- Grave funções para todas as transformações de dados que você aplicar, por cinco motivos:
    - Assim, você pode preparar facilmente os dados da próxima vez que obtiver um novo conjunto de dados
    - Então, você pode aplicar essas transformações em projetos futuros
    - Para limpar e preparar o conjunto de teste
    - Para limpar e preparar novas instâncias de dados assim que sua solução estiver ativa
    - Para facilitar o tratamento de suas escolhas de preparação como hiperparâmetros

---

1. Limpeza de dados:
    - Corrija ou remova outliers (opcional).
    - Preencha os valores ausentes (por exemplo, com zero, média, mediana ...) ou elimine suas linhas (ou colunas).
2. Seleção de features (opcional):
    - Elimine os atributos que não fornecem informações úteis para a tarefa.

3. Engenharia de features, quando apropriado:
    - Discretize recursos contínuos.
    - Decompor recursos (por exemplo, categórico, data / hora etc.).
    - Adicione transformações promissoras de recursos (por exemplo, log (x), sqrt (x), x2, etc.).
    - Agregue recursos em novos recursos promissores.
4. Escalonamento de recursos:
    - Padronize ou normalize recursos.

5. Explore muitos modelos diferentes e selecione os melhores.

Notas:

- Se os dados forem enormes, você pode querer amostrar conjuntos de treinamento menores para que possa treinar muitos modelos diferentes em um tempo razoável (esteja ciente de que isso penaliza modelos complexos, como grandes redes neurais ou florestas aleatórias).
- Mais uma vez, tente automatizar essas etapas o máximo possível.

---

1. Treine modelos de diferentes categorias (por exemplo, linear, Nayve Bayes, SVM, Random Forest, Redes Neural, etc.) usando parâmetros padrão.
2. Avalie e compare seu desempenho.
    - Para cada modelo, use a validação cruzada N-fold e calcule a média e o desvio padrão da medida de desempenho nas N dobras.
3. Analise as variáveis mais significativas para cada algoritmo.
4. Analise os tipos de erros que os modelos cometem.
    - Quais dados um humano teria usado para evitar esses erros?
5. Execute uma rodada rápida de seleção e engenharia de features.
6. Execute mais uma ou duas iterações rápidas das cinco etapas anteriores.
7. Lista os três a cinco modelos mais promissores, preferindo modelos que cometem diferentes tipos de erros.

6. Ajuste seus modelos e combine-os em uma ótima solução.

Notas:

- Você desejará usar o máximo de dados possível para esta etapa, especialmente à medida que avança para o final do ajuste fino.
- Como sempre, automatize o que puder.

---

1. Ajuste os hiperparâmetros usando validação cruzada:
    - Trate suas escolhas de transformação de dados como hiperparâmetros, especialmente quando você não tem certeza sobre eles (por exemplo, se você não tem certeza se substitui os valores ausentes por zeros ou pelo valor mediano, ou apenas descarta as linhas).
    - A menos que haja poucos valores de hiperparâmetros para explorar, prefira a pesquisa aleatória em vez da pesquisa em grade. Se o treinamento for muito longo, você pode preferir uma abordagem de otimização Bayesiana.
2. Experimente os métodos do Ensemble. Combinar seus melhores modelos geralmente produzirá melhor desempenho do que executá-los individualmente.

3. Assim que estiver confiante sobre seu modelo final, meça seu desempenho no conjunto de testes para estimar o erro de generalização.

> **AVISO**
>
> **Não ajuste seu modelo depois de medir o erro de generalização: você apenas começaria a ajustar o conjunto de teste.**

7. Apresente sua solução.

1. Documente o que você fez.
2. Crie uma boa apresentação.
   - Certifique-se de destacar o quadro geral primeiro.
3. Explique por que sua solução atinge o objetivo de negócios.
4. Não se esqueça de apresentar pontos interessantes que você notou ao longo do caminho.
   - Descreva o que funcionou e o que não funcionou.
   - Liste suas suposições e as limitações do seu sistema.
5. Certifique-se de que suas principais descobertas sejam comunicadas por meio de belas visualizações ou declarações fáceis de lembrar (por exemplo, "a renda média é o indicador número um dos preços das moradias").

8. Coloquei seu projeto em produção!

1. Prepare sua solução para produção (conecte-a às entradas de dados de produção, escreva testes de unidade, etc.).
2. Escreva o código de monitoramento para verificar o desempenho ao vivo do seu sistema em intervalos regulares e acione alertas quando ele cair.
   - Cuidado com a degradação lenta: os modelos tendem a “apodrecer” conforme os dados evoluem.
   - A medição do desempenho pode exigir um pipeline humano (por exemplo, por meio de um serviço de crowdsourcing).
   - Monitore também a qualidade de suas entradas (por exemplo, um sensor com defeito enviando valores aleatórios ou a saída de outra equipe se tornando obsoleta). Isso é particularmente importante para sistemas de aprendizagem online.
3. Treine seus modelos regularmente com base em dados atualizados (automatize o máximo possível).