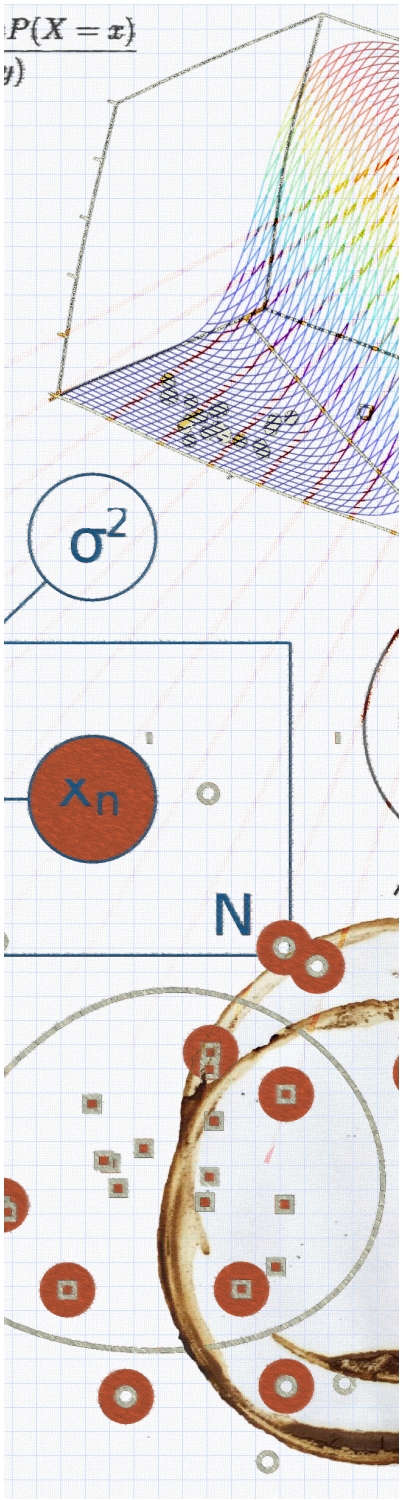
Machine Learning Guide for Policing




# RUDI

## The full framework

Ruth Spence, Tamara Polajnar, Hazel Sayer
STAR 2023/2024

## Contents

# Purpose of RUDI

While there is a proliferation of algorithm use in policing and academic advice for ethical outcomes of algorithm use, there is little practical guidance for implementing algorithms with direct effect on members of the public. RUDI (which stands for Rationale, Unification, Development, Implementation) is a flexible framework that can be adapted to different types of algorithmic modelling within police departments. It provides a structured guide that can ensure fairness and transparency. RUDI can also foster better communication across forces due to the use of standardised documents.

While we recommend forces develop internal capacity for modelling, outsourcing should not limit transparency or adherence to this framework.

An ethical framework for developing algorithms, such as RUDI, is essential in the context of policing, because it ensures:

### Fairness and Equity

An ethical framework ensures that algorithms are designed and implemented with a commitment to fairness and equity. Mitigating biases and striving for unbiased decision-making fosters a more just and equitable system.

### Transparency and Accountability

Transparency fosters accountability and is vital for building trust in algorithmic decision-making processes. Encouraging the development of algorithms that are transparent and explainable allows stakeholders to understand how decisions are reached.

### Community Trust and Engagement

Involving community stakeholders in the development and deployment of algorithms fosters trust between law enforcement and the public.

### Mitigation of Bias

Biases in data and algorithms can lead to discriminatory outcomes. Proactive steps should be taken to identify, address, and mitigate biases in both data and algorithms. This ensures that law enforcement technologies do not disproportionately impact certain demographics or communities.

### Legal Compliance

People can challenge the results of the model, adhering to legal standards and good documentation is essential for ensuring law enforcement agencies can justify and explain their decision-making process.

### Continuous Improvement and Adaptation

Continuous monitoring, auditing, and impact assessments, allow for the identification of ethical concerns and the implementation of improvements over time.

### Prevention of Unintended Consequences

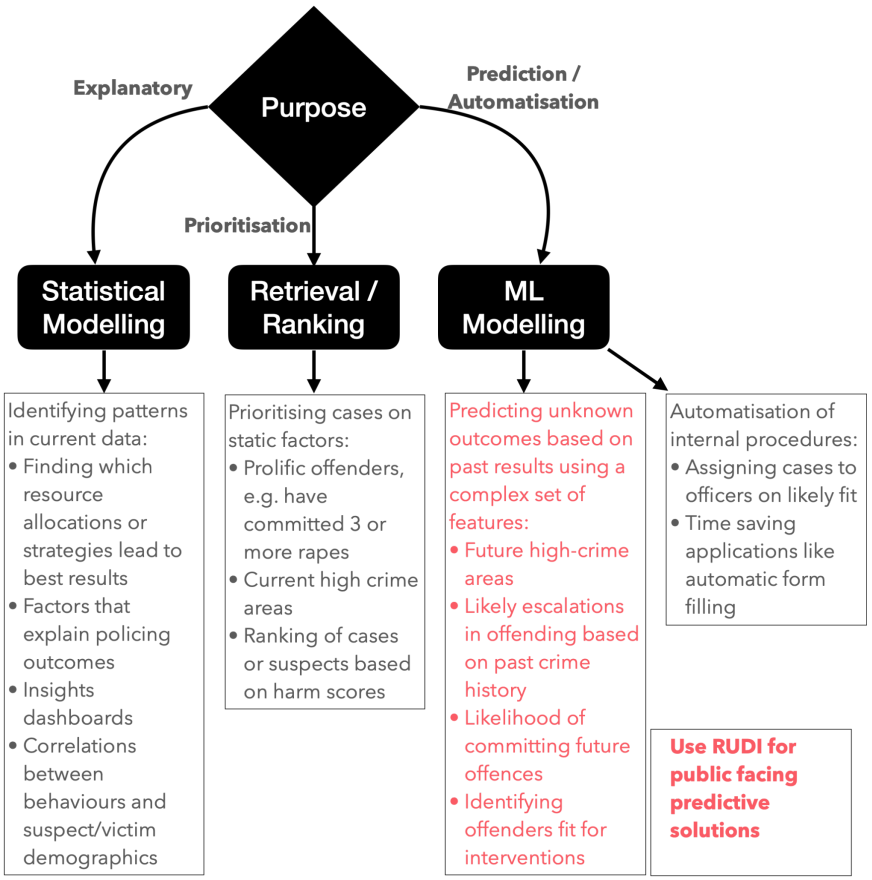
Considering potential unintended consequences of algorithmic decision-making can help prevent negative consequences and ensures that technology is used responsibly.

RUDI serves as a guide for ethical and accountable development and deployment of algorithms, promoting transparency and responsible practice.

# What is Modelling

There are different ways of leveraging data and knowledge collected by an organisation. There are roughly three ways statistics can be used to leverage data:

1. Explanatory statistical modelling: The way it is usually applied by criminologists explanatory modelling refers to "the application of statistical models to data for testing causal hypotheses about theoretical constructs"[1]. For example, using statistics to discover which recorded behaviours are correlated with multiple violent crimes. This does not involve predicting outcomes for new sets of behaviours, but rather finding out which features are associated, and to which degree, with outcomes.
2. Prioritisation through retrieval or ranking: Using a database search or a search engine to pick out data entries that conform to a set of specified factors to rank or prioritise results. For example, ranking suspects in the area by cumulative Cambridge Crime Harm Index (CCHI), or finding nominals who have been suspected of three or more rapes that have yet to be charged and prioritising their cases for interventions and investigation. This also does not involve prediction or creation of any new features or information.
3. Prediction or automatisation through machine learning: Using data to train a model to embody patterns in the data and produce new information, such as labels, or text, or predictions of likely future events. Special care needs to be taken when using predictive modelling in decision making and record augmentation, especially when concerning individuals, communities, or geographies.




It can be useful to examine using explanatory modelling and retrieval as potential solutions, as they directly leverage knowledge within the contained data in a more transparent and explainable way, before attempting other forms of modelling. The techniques for explanatory and predictive modelling are not mutually exclusive, and familiarity and insights gathered from examining patterns in data can inform

[1] https://www.stat.berkeley.edu/~aldous/157/Papers/shmueli.pdf

predictive modelling and vice versa. The remainder of this document is mainly concerned with predictive modelling using ML for the cases which have direct impact on individuals or communities.

# Prior to Predictive Modelling

Introducing modelling is a commitment both in terms of time and resources. To successfully develop and implement algorithmic modelling in policing, diverse stakeholders must work together to define why and how modelling will work in practice, as well as have the specific expertise needed to develop and test large data models. In lieu of a centralised body that can leverage machine learning techniques on a national level, we recommend police develop and build their own models in-house for two reasons: firstly, the model will be specific to the aims and data of each force and secondly, any model needs to be maintained over time.

### Conceptualisation

We recommend police consider what they want to achieve through data modelling. Although guidance on how to conceptualise problems suitable for modelling is outside the scope of this framework, before beginning data modelling forces should be able to answer the following:

### Conceptualisation template

| | |
|---|---|
| What is the problem to be addressed? | |
| What is the proposed solution? | |
| What is the overall aim of the initiative and why is this important? | |
| Who/what does the initiative target and why? | |
| What are the key definitions being used and how are they operationalised? (e.g., high harm, recidivism risk) | |
| What is the mechanism (e.g. professional judgement, structured professional judgement using a tool, static algorithm, machine-learning based algorithm) by which cases of interest will be identified and why? | |
| Will identified cases go through further assessment? If so, what does this look like? | |
| What kind of action will be taken? How is this justified and is there a resource available for this? | |
| How does all of the above fit within the legal and ethical frameworks in which policing operates? | |
| What evidence base are you using to justify all of the above steps? (briefly state underlying theory, hypotheses, or evidence from prototyping) | |

*Note: We suggest you do not proceed any further until you can complete the above table*

## RUDI

RUDI is a framework police forces can follow to build and implement data modelling. It is specific to complex machine learning models which can drive police action and have potential public impact, such as models that prioritise individuals or predict future criminal behaviour or locations. It covers four main stages:

Rationale: Documenting the process, making decisions explicit and justifying actions during unification, development and implementation.

Unification: Merge data sources together for modelling and ensure validity and reliability of data.

Development: Build and test models, evaluating for bias, performance and limitations and choose the preferred model.

Implementation: How the model feeds into current practice and how it will be maintained over time.

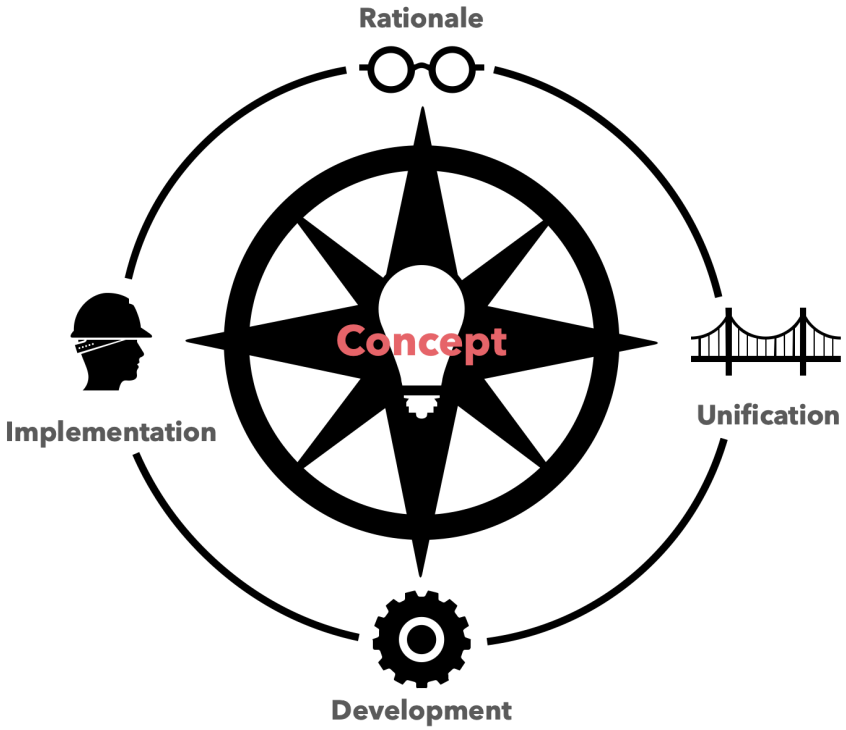



Figure 1: RUDI process

Project evolution is not a linear process (see Figure 1); however, conceptualisation of the problem always comes first, and implementation of the proposed solution always comes last. Rationale, unification, and development all pose reasonable entry points into project execution, and throughout the project lifecycle, the team will have to revisit each of these stages until an acceptable solution is found for the problems outlined at the conceptualisation stage. Implementation includes model monitoring and updates which will necessitate a return to the earlier stages as well.

There is no 'one right way' to develop data models, it is a series of decisions that must be made whilst balancing multiple competing concerns. RUDI is a framework that sets out some of the decisions involved and provides a way for forces to document the process to improve transparency (being honest and open about decisions and trade-offs), justifiability (making decisions defensible), lawfulness (using proportionate methods and data) and accountability (being answerable to critique). It does not prescribe an exact formula for algorithmic modelling, nor does it reduce the need for in-house domain or data expertise (see Data Scientist Expertise for details of data expertise required).

# Rationale

## Responsible team members: The whole team

### Assign a team and a senior reporting officer to ensure full accountability of the tool's performance and its deployment

- Multi-disciplinary teams:
    - Enable you to apply different expertise to the problem and gather input on ethical considerations and potential impacts.
    - Enable internal auditing i.e., the person developing the model is different from the person reviewing the model's statistical performance.
    - Enable external auditing i.e., external scrutiny of the process, especially for high-risk[2] models.

You may not have personnel available to fill all the separate positions, if so, think through the implications of this. For example, using an external agency to fulfil all these roles may lead to problems when maintaining the model.

- Effective multidisciplinary working:
    - Develop a shared understanding of role responsibilities, tasks and procedures involved
    - Have regular team meetings where working relationships are actively developed and good communication practices established (e.g., minuted meetings)

Senior Reporting Officer: Takes strategic and managerial responsibility of the project, including delivery and ongoing evaluation.

Domain Experts: Possess subject matter expertise and will use the model for their work, set out the problem the model is being used to solve and what the required outcomes are. Help evaluate if the model is achieving the intended outcomes.

Data Engineers: Responsible for the data used by the models, they unify and standardise the data for use in the model.

Data Scientists: Create, evaluate and maintain models, including associated documentation

Validators: Review and evaluate the work of the data engineers and scientists, with a focus on technical accuracy. Oftentimes, validators are data scientists who are not associated with the specific model or project at hand.

Governance Personnel: Review and approve the work created by both data engineers and data scientists, with a focus on risk.

Validators and governance personnel can include ethics boards, academics and community stakeholders who are external to the force and can review the work.

### Create a clear business case

- Document all decisions so the entire process is transparent and auditable
- Document trade-offs e.g., increased accuracy often means decreased explicability

[2] You may consider formalising the risk level of the the project using a risk register
Risk Assessment Questionnaire | Interpol

- Discuss the business case before any modelling takes place
- Modelling is an iterative and non-linear process – sections of the business case will change as the project progresses; this should be documented along with the reasons for change

### Cost the computational resources

Training and using some types of machine learning models may require computational resources that are only available to you through cloud computing. This may be costly. Make sure you adequately calculate the potential development and hosting costs for the final models. It may be possible to reduce costs of deep learning by using computational tricks such as quantisation and adapters to train smaller versions of more powerful models.

### Business case template

| | |
|---|---|
| Team | |
| Senior Responsible Officer | |
| Data Engineer(s) | |
| Data Scientist(s) | |
| Domain Expert(s) | |
| Validator(s) | |
| Governance Expert(s) | |
| Plan for if someone leaves post | |
| | |
| Model | |
| Outline the problem to be addressed and the overall aim of the model, including if relevant, who the model is being used to target, and why | |
| Why is algorithmic modelling rather than other options (e.g., professional judgement) best suited to solving this problem? | |
| Alignment with force priorities | |
| Alignment with national strategic priorities | |
| Briefly state underlying theory, hypotheses, or evidence from prototyping | |
| Desired outcome(s) i.e., how will you know the model is working? | |
| Possible undesired outcome(s) (e.g., either directly or through misuse) | |
| Model design (e.g., classification, ranking) & rationale | |

| | |
|---|---|
| Data features needed & rationale | |
| Data analysis plan & rationale (e.g., how bias will be assessed, how the model will be evaluated and analysis of errors) | |
| Inclusion and exclusion criteria for cases being included in the model & rationale | |
| Plan for storing & sharing output | |
| Who will use the output & will training on using the output be provided? If yes what? And if no, why not? | |
| How will the model's outputs be incorporated into officer decision-making? Are processes in place to keep track of accuracy and to catch model drift? | |
| If relevant, what is the intervention for the cases the model identifies? Are there situations where identified cases will not be considered for an intervention? | |
| What are the implications of an error (both false positives and false negatives) e.g., ethical, legal, reputational? | |
| What is the plan for ongoing model evaluation (e.g., thresholds and inputs/outputs that trigger model retraining)? | |
| Can iterative changes be made to the model as needed? (e.g., to account for feedback loops due to the effects of interventions) | |
| | |
| Costs & Resourcing | |
| What costs/resources will be needed for set up & piloting? | |
| What costs/resources will be needed for model maintenance? | |
| | |
| Changes & Trade-offs (to be completed during project lifecycle) | |
| Any changes to model design & rationale | |
| Any changes to analysis plan & rationale | |

| | |
|---|---|
| Any trade-offs & rationale e.g., false positive/negative rates, accuracy vs. explicability | |

*Note: this is not exhaustive and should be added to in line with the force's own concerns. Responses will change as the project progresses*

# Unification

## Responsible team members: Data Engineer and Data Scientist to lead

### Define what data is needed, which system(s) it is stored in and if you have software to analyse it

- Identify what data features are needed. These should be clearly justified by specific aims and requirements defined beforehand using Rationale and a Data Protection Impact Assessment (DPIA).
- Ensure features can be accessed. This may involve discussions with third party system owners (e.g., Connect, Niche) about being able to extract data.
- Ensure you have correct software for analysis. This may involve discussions with your IT department.

If using external agencies: personally identifiable information is subject to regulation governing its collection and use; therefore, pseudonymise data before sharing.

### Build integrated system using 'match and merge process'

- Build tables and link them together using standardised keys to reduce data anomalies and make the data easier to analyse
    - Extract relevant features from across the data
    - Merge records under one identifying variable (e.g., suspect ID) by using overlapping data features (e.g., same DOB, same address)
    - String comparison measures can measure how close near-matches are
    - Working out the links across different records can take a lot of manual work

A lot of information is included in written text so it can involve a lot of feature engineering and be computationally expensive.

Pulling out data from backend databases can slow down the operational use of systems and so must be balanced with other needs.

### Quality-assess the system

- Conduct cross-checks to ensure the accuracy and validity of the data as data inaccuracies can lead to mistaken identity or inaccurate intelligence influencing downstream decisions.
- Cross-checks can feedback into further match and merge processes e.g., overlaps on one identifying variable may help merge records elsewhere.
- Conduct unit testing to ensure the code components work as expected.

### Set up Extract Transform Load (ETL) code to run on a regular basis (whatever frequency you decide)

- Ensure ETL code has quality checks included so new data and data formats are caught.

### Run business reports

- Business reports can be used to summarise, track and analyse patterns in the data, including trends and irregularities.
- At this point there may be enough available insight that further modelling is not needed.

# Development

## Responsible team members: Data Scientist to lead

This section is intended as indicative of the steps involved in analysis not an exhaustive list of what analysis to conduct

### Project Preparation

It is important to take the time in the beginning of the project to set up data and code versioning. As you explore the solution space including various iterations of features and models, it will be important to keep track of the steps that led to the best performing solutions.

### Data Preparation (Preprocessing)

These steps will have to be replicated to run the model in production so ensure that all processes are automated and clearly documented. Since data preprocessing can be iterative through the project lifecycle, use versioning to link different data and code versions to the models and results to preserve replicability and keep track of the best models.

- Importing: Write code that gathers dataset features from the available unified data and verifies data integrity based on any assumptions that are made on formats, values, or quality.
- Cleaning and feature manipulation: Remove or correct entries so the data is valid and reliable. This includes removing cases that are inappropriate, as outlined in the rationale, as well as assessing missing data. You may also end up scaling or combining features, so you need to pay special attention to how scaling factors translate to unseen examples.
- Labelling: Accurate data labelling improves model predictions. Labels should be consistently applied based on the aim of the model using predetermined criteria, so the labels reflect this as closely as possible (e.g., high risk cases are labelled as such if they meet certain criterion, such as having a particular harm score or a certain number of convictions).
- Assess and mitigate bias: Try to discover and control the biases before using the data to train models
    - Where it is not practical to remove bias, quantify and minimise it where possible.
    - The team should decide what is the appropriate quantitative and qualitative definition of bias to determine what is acceptable error, e.g., what is the allowable difference in error rate for different groups.

### Splitting Data

- Split your data into training, validation and test data
    - Training data: used to develop the model
    - Validation data: used to verify if the chosen model works correctly
    - Test data: kept aside to ensure that the model does not overfit to training data

- If there is not much data available cross-validation can be conducted on the training set

### Modelling

Train the models on the training data and conduct error analysis on the validation data:

- Any trade-offs between accuracy and interpretability should be clearly thought through and documented
    - Evaluate how good each model is for achieving the aim based on pre-set criteria
    - Choose the best model and investigate error rates, sources of error, assess for bias and limitations within the model, check model biases against the underlying biases in the data.
    - Consider different misclassification costs and consider the asymmetry of misclassification costs (e.g., in risk assessment, incarceration versus wasted police time)
    - If you uncover patterns in the sources of error or unfairness, go back through feature engineering, data splitting, and modelling to improve the process
- Do final checks on the held-out test data
    - Check for bias on the chosen model, document the trade-offs
    - Conduct interpretability/explicability tests on the chosen model, record any known patterns

### Documentation

- Enables other people to understand the dataset and the process through which the model was developed:
    - Data cards: supply a quick look at the properties of the dataset
    - Model cards: document processes so models can be compared and evaluated

### Template Data Card

| Metadata | |
|---|---|
| Senior Reporting Officer | |
| File name | |
| File format | |
| Short summary of dataset purpose and content, and keywords | |
| Dataset size | |
| Version history | |
| Data governance plan | |
| % of missing values for relevant variables, nature of missingness, causes (if known) and how missing data has been handled | |
| Data Sources | |
| List sources of data and why each was included | |
| Variables | |

| | |
|---|---|
| Description of each variable (column) in the dataset | |
| Data Preparation | |
| How was the data sampled from the source data and how was it split into training/validation/test sets? | |
| Describe what data cleaning took place e.g., cases removed, discretisation, coding, changes etc. | |
| Describe data labels and what criteria were used for each? | |
| How was missingness assessed? | |
| Evaluation Data | |
| What data was used to train and test the model? Why? | |

## Template Model Card

| | |
|---|---|
| Model Details | |
| Senior Responsible Officer | |
| File name | |
| Date & Model Version | |
| Type of model (e.g., random forest, neural net) | |
| Intended Use (can be taken from business case) | |
| Aim | |
| Who will use the output | |
| Appropriate & Inappropriate case examples | |
| Results | |
| What groups are in the data? Why have these specific groups been chosen and how were they defined? | |
| Are there any known groups missing from the data? Why might this be? | |
| Unitary results (i.e., by group), including uncertainty values | |
| Intersectional results (i.e., where groups are combined), including uncertainty values | |

| | |
|---|---|
| Error Analysis | |
| What groups are in the data for which model performance might vary? (e.g., ethnicity, location, crime type) | |
| Have you reported differences for all relevant features? If not, why | |
| What has been done to mitigate any bias? | |
| Metrics | |
| What are the model's performance measures, including why these were selected as appropriate (e.g., false positive/negatives, distribution differences across groups) | |
| What are the decision thresholds (if relevant) including what they are and why they were chosen | |
| How were uncertainty and variability calculated? (e.g., variance, confidence intervals) | |
| Risks | |
| What risks might be present in model use (e.g., recipients, likelihood and magnitude of possible harm) | |
| What risk mitigation strategies were used during development? | |
| Maintenance | |
| Any known or expected changes relating to the population or how data were collected which may contribute to data shifts over time? | |
| What metrics will trigger an update and at what thresholds*? By whom and how will changes be communicated? | |

Note: *thresholds are not relevant to time series or spatio-temporal models*

# Implementation

## Responsible team members: The whole team

### Fitting model into current practice

- Work closely with relevant personnel and decision-makers to understand the specific needs and challenges of the operational environment so deployment aligns with workflows.
- Have a data literacy strategy that includes training on:

    - Processes: who will access the model, how will they access and interpret the model results, limitations of the model, what to do if the model agrees/disagrees with end user assessment, relevant interventions etc.
    - Good data entry: modelling is less effective with unreliable, missing or invalid data
    - Confirmation bias:[3] the tendency of people to look for information that confirms or strengthens their beliefs and overlook information that challenges or contradicts them. It is difficult to dislodge and can affect decision making if you ignore or do not seek out information that may contradict your initial assessment. Use "consider the opposite"[4] strategy.
    - Automation bias:[5] the tendency to automatically defer to the algorithm's result, despite contradictory information. This is most likely to occur when the model is seen as highly accurate and reliable and therefore may become more problematic as staff become more accustomed to using algorithmic outputs. Exposing staff to failures during training can guard against complacency, whereas just telling them about the limitations and warning them to always verify does not sufficiently reduce automation bias. Nevertheless, staff should be made aware of the limitations of the model, its failure rate and the importance of keeping a "human in the loop" because of their ability to consider context and their legal requirement to hold ultimate responsibility for any decisions made.
    - Model Misuse: Model misuse occurs when a machine learning model is applied for purposes beyond the scope of its original training objectives. This means that you have to use the model for the purpose it was trained for and as close to the way it was trained as possible. For example, if a model was trained on robbery crime histories and was trained to recognise people who are likely to escalate and commit violent offences, you cannot apply it to other types of crimes, or for detection of more moderate offences without retraining.
- Develop user-friendly interfaces for staff to interact with the model outputs. Consider incorporating feedback mechanisms
    - Usability testing:[6] can reveal previously unknown user experience problems.
    - A/B testing:[7] the user is randomly given either design A or design B, both of which differ in one specific way (e.g., how the output is presented or when the output is presented etc.). The highest performing variation can then be identified according to predefined metrics to make data-driven design decisions.
- Feedback mechanisms should be used on an ongoing basis:
    - Feedback from staff can help determine the best way to implement modelling in practice.
    - Feedback from community stakeholders can help ensure modelling is not having undue negative impacts on different groups

- Capacity: Additional resources are necessary to support the implementation of algorithms, both in terms of developing and maintaining the models but also to provide a full and appropriate response to algorithmic predictions

[3] Confirmation Bias | Ethics Defined

[4] Consider the Opposite - A Debiasing Strategy

[5] What is Automation Bias?

[6] Usability testing: qualitative studies - GOV.UK

[7] A/B testing: comparative studies - GOV.UK

## Implementation decisions

| | |
|---|---|
| Who will access the model results? | |
| What training is provided to staff accessing model results? What provision is in place to ensure relevant staff get this training? | |
| At what point in the investigative process will staff access the model results? | |
| How will model results be presented? | |
| How will model limitations be presented? | |
| When should predictions be acted upon? What should staff do? | |
| Are there occasions where predictions should not be acted on? If so, when? | |
| How do staff document why they have either agreed/disagreed with the model's predictions? Can they document contradictory evidence? | |
| What additional resources will be needed to act on the model's predictions? | |
| Are there processes in place for when the model is clearly making incorrect or biased decisions? | |

## Ensuring the Model Works in Practice: Data Scientist to lead

- Shadowing
    - The model is deployed alongside existing practice to test its performance and investigate how it compares to current procedures. This process should last as long as necessary to enable the accuracy of the model's predictions with real-world data to be assessed (i.e., if it predicts risk over 12 months it needs to run for 12 months). The accuracy can then also be compared to current practice.
- Ongoing monitoring: regularly analyse model outputs and assess for any deviations or degradation in performance that may indicate the need for retraining or adjustment.
    - Data drift: The underlying data distribution changes, which changes the relationship between the input and the output and reduces accuracy. This can be gradual or can be sudden (e.g., counting rule changes)
    - Concept drift: the relationship between your input and what you want to predict changes
    - Managing Drift
        - Maintain baseline models for performance comparisons
        - Regularly retrain and update models
        - Evaluate the importance of new data
        - Develop a monitoring dashboard and tools to track model performance in production
            - decide on metrics you are going to track to guard against data drift and concept drift and set threshold alarms. These can be based on numerous metrics depending on what is most relevant

    - metrics and thresholds can also change over time and therefore these also need to be monitored
- Plan for feedback loops caused by interventions
- Plan what to do if the model no longer works satisfactorily i.e., whilst the model is being retrained
- Develop criteria to determine how you know the modelling is making a difference i.e., what outcome is expected as part of your rationale.
- Impact Assessments: Regularly audit the model's outputs across different groups to ensure fairness and prevent unintended consequences.

## Model Maintenance

| | |
|---|---|
| What metrics will be used to track model performance and why have these been chosen? | |
| What thresholds for each metric will trigger an alarm and why have these been chosen? | |
| How many thresholds need to be breached for the model's performance to be reassessed? Why? | |
| How will the model's performance be reassessed? | |
| What will happen if the model's performance needs to be reassessed? (e.g., stop using the model entirely? warn users the output may be faulty?) | |
| What additional resources will be needed to maintain the model? | |
| What is the process in place for assessing the impact on different community groups? | |

Note: *thresholds are excluded for time series or spatio-temporal models*

# Appendices

## Key Concepts and Definitions

- Model: A model in machine learning refers to the mathematical representation of a real-world process or system. It is created based on training data and is designed to make predictions or decisions without explicit programming.
- Algorithm: An algorithm is a set of step-by-step instructions or rules designed to solve a specific problem or perform a particular task. In the context of machine learning, algorithms are utilised to train models and make predictions.
- Model Misuse: Model misuse occurs when a machine learning model is applied for purposes beyond the scope of its original training objectives. This deviation from the intended application poses risks and challenges, as the model may lack the necessary understanding or context to make accurate and reliable predictions in unfamiliar domains. Model misuse is considered undesirable because it can lead to erroneous outcomes, compromised accuracy, and potential ethical or legal implications. Using a model outside its designated scope underscores the importance of thoughtful consideration and understanding of the model's limitations and capabilities.
- Training data: Training data consists of examples used to train a machine learning model. It includes input features and corresponding target variables, enabling the model to learn patterns and relationships.
- Evaluation metrics: Evaluation metrics are quantitative measures used to assess the performance of a machine learning model. Common metrics include accuracy, precision, recall, and F1 score, depending on the specific goals and characteristics of the problem at hand.
- Feature Engineering: Feature engineering involves selecting, transforming, or creating relevant input features for a machine learning model. Well-crafted features enhance the model's ability to capture meaningful patterns in the data.
- Overfitting: Overfitting occurs when a machine learning model performs well on the training data but fails to generalise to new, unseen data. It is a result of the model learning noise in the training set rather than the underlying patterns.
- Deployment: Deployment refers to the process of integrating a trained machine learning model into a production environment, making it accessible for making real-time predictions or decisions.
- Production Environment: The production environment is the live, operational system where a machine learning model is deployed to make real-time predictions or decisions. It is the environment in which the model interacts with end-users or other systems.
- Data Drift: Data drift refers to the phenomenon in machine learning where the statistical properties of the input data used for training a model change over time in the production environment. This can happen easily within the policing context as policies, input procedures, and effects of model use lead to differences between training data and the data being fed into the live system. These changes can adversely impact the model's performance as it may no longer accurately represent the underlying patterns in the evolving dataset. Monitoring and addressing data drift is essential to maintaining the effectiveness and reliability of machine learning models deployed in dynamic and changing environments.
- Concept drift: Concept drift occurs when the statistical properties of the target variable (the phenomenon to be predicted) change over time, leading to a decline in the model's performance. It is crucial to monitor and adapt models to account for concept drift in dynamic environments.

# Further Notes on Development for Data Scientists

This section provides a more detailed guide for developing machine learning (ML) models intended for use in policing. It's not a complete list of analyses to conduct. Whether you're new to machine learning, have academic experience but little exposure to deploying AI in human-centric environments, or you're an expert, here are some reminders about the steps involved in building a model-based product. Towards the end of the document, you'll find a list of useful resources, including links to online courses that cover much of this material. They also provide general advice on building and deploying ML models.

Note: Many of the suggestions in these guidelines are based on python. This is based on the proliferation of open-source code for machine learning rather than a prescriptive choice of language.

## Project Preparation

Many projects begin their journey as prototypes, and while proof of concept is essential, the transition to a deployable product with real-world impact requires: a) adherence to sound coding and engineering practices, and b) the establishment of a reproducible and well-documented process, even if it demands additional time investment.

Creating a well-organised work environment with traceable code and dataset modifications is pivotal to achieving these objectives.[8] Resources like DVC's best practices[9] offer insightful guidelines, or you can explore comprehensive solutions such as MLFlow[10], providing local management capabilities and seamless integration with diverse cloud computing platforms. Alternatively, if a single provider is your preference, platforms like Azure Pipelines[11] offer integrated solutions.

Each machine learning project involves a collection of decisions, some of which are testable, and others less tangible. Even seemingly trivial choices, such as row normalisation versus column normalisation, or the selection of a model family, can significantly impact model effectiveness. The risk of coding errors and faulty assumptions is heightened when working in isolation. We strongly recommend collaboration with other data scientists. Regular discussions within interdisciplinary teams about assumptions and decisions that influence the final product can enhance the overall quality of the project.

A deep understanding of the dataset, including its contents, errors, and the underlying police procedures and processes that contribute to its creation, is essential. Recognising inherent biases is crucial for informed decision-making and selecting appropriate modelling techniques. Automatic data visualisation packages and automated ML training can be beneficial in gaining insights. Techniques such as data clustering can unveil patterns, assess linear separability, and guide automated labelling processes towards promising directions.

---

[8] #23 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 15]

[9] Versioning Data and Models | Data Version Control · DVC

[10] What is MLflow?

[11] Creating End-to-End MLOps pipelines using Azure ML and Azure Pipelines – Part 1

# Data Preparation

Data profiling[12] and visualisation[13] tools can illuminate insights and facilitate comprehensive understanding during the decision-making process. Here is a non-exhaustive guide to data preparation:

### Create a unified dataset by importing relevant data from various sources

- Check your data at each point in pre-processing.
    - Perform rigorous checks at each stage of pre-processing, especially if the unification code is developed by another party.
    - Carefully address potential errors and misalignments resulting from data manipulations.
    - It's easy to inadvertently introduce features that mirror labels, particularly if labels are auto-generated from the data.
    - Regularly assess the impact of data manipulations on model performance.

### Cleaning and feature manipulation

The presence of incorrect or inconsistent data can distort the model's results. Data cleaning involves removing or correcting entries, so the data is valid and reliable. This includes removing cases that are inappropriate, as outlined in the rationale.

- Correct data as necessary:
    - Tabular data can have a lot of errors in it because of manual entry and various changes to the input systems. You should decide what to do for each column based on types. Some examples include:
        - Unlikely age values less than 0 or bigger than 100
        - Dates that are unlikely e.g. 1900
        - Values that can be propagated when they are available for other entries for the same person, such as gender.
        - Columns with single or too few values should be dropped. While there are imputation techniques, you need to be careful about the implications for your data, as imputation in some cases might be equivalent to introducing false facts.
        - Only some ML models can use null values, if using data preprocessing pipelines like those available in sklearn[14], you can delay decisions and apply transformations on a case-by-case basis.
        - Correctly encode and process categorical features and ordinal features. Again, data processing pipelines can help you process each of these separately and customise input for different models.
        - The efficient way to improve the accuracy of AI model: Andrew Ng's Data-centric AI | by Dasol Hong | AI Network | Medium

---

[12] These can be free and easy to use like YData Profiling , although there are lots that are not code based 16 Open Source Data Profiling Tools (Plus Benefits) | Indeed.com

[13] https://towardsdatascience.com/visualizing-geospatial-data-in-python-e070374fe621
https://www.datacamp.com/tutorial/geospatial-data-python
https://machinelearningmastery.com/data-visualization-in-python-with-matplotlib-seaborn-and-bokeh/
A comprehensive and practical guide to Image Processing and Computer Vision using Python: Part 1 (Introduction) | by Pranav Natekar | Towards Data Science

[14] https://scikit-learn.org/stable/modules/preprocessing.html
Nice example of pipelines for use in the stacking classifier with tree-based and linear methods
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.StackingClassifier.html

- Sequence/Geospatial
  - Standardise time variables and consider treating parts of the timestamp as separate features.
  - Unlike tabular data, sequence data may require imputation of missing values[15]
  - Features may need to be encoded or transformed to make them more standard or useful
  - Some sequence processing examples: https://towardsdatascience.com/preprocessing-time-series-data-for-supervised-learning-2e27493f44ae
  - Leverage distance measures and shape processing for geospatial data.
  - Some geospatial examples: https://towardsdatascience.com/geopandas-hands-on-building-geospatial-machine-learning-pipeline-9ea8ae276a15
- Image/Video
  - Data Cleaning for Image Classification | by Aidan Coco
  - https://medium.com/unpackai/common-data-cleansing-problems-and-approaches-for-image-classification-models-1276ff4390f3
  - Video Preprocessor and Augmentation for Deep Learning tasks | by Biplab Barman | Analytics Vidhya | Medium
- Text data processing is a rapidly changing field. Many techniques that apply to older models are unnecessary if you are using deep-learning-based methods. Treating text data as categorical so it can be used in models suitable for tabular data like SVMs or trees leads to word meaning in context being disregarded. However, you can combine tabular and text data by generating the text vectors separately, for example, by using a model like BERT.
  - If you are using single tokens or n-grams, consider a) removing very high and very low frequency words; b) using stemming; c) l2-normalising by row, not by column; d) using methods such as PPMI to reweigh the features; e) applying SVD or LDA to smooth over similar words; etc. A combination of PPMI, l2-norm, followed by SVD can be very useful[16].

- Some manipulations depend on the statistical distribution of known data, such as statistics-based normalisations and data scaling. Take care to analyse how these will be done on incoming cases as you might need to:
  - Estimate them only on the training portion of your data
  - Save any scaling factors so you can apply them to the test/production data
- The effects of missing data depend on the task, but it's important to test what is the best strategy for dealing with your data. This may be tricky as you can end up changing both your training and test sets, making comparison of the methods more difficult. Keep track of issues and performance. Two main ways of dealing with missing data are deletion and imputation, you can delete bad features or bad training examples, but you might be missing information. You can estimate the missing data using imputation, collaborative filtering, or dimensionality reduction techniques such as singular value decomposition; however, you need to be careful not to introduce erroneous information. Likewise, you need to think about how you will deal with the missing data when your system is running in production.
  You might be able to find an algorithm that suits your data because it handles the missing values in

[15] 6.4. Imputation of missing values — scikit-learn 1.4.0 documentation

[16] Improving Distributional Semantic Vectors through Context Selection and Normalisation

a particular way[17] or you might find that knowledge of how an algorithm works can tell you the best way to represent the missing values. For example, using a 0 might be mistaken for genuine 0 values, as can something like –1, so you might be better off using a very unlikely value like –999 if you are using tree-based methods. On the other hand, if you are normalising, this can cause you issues.

- How to Handle Missing Data with Python - MachineLearningMastery.com
- 7 Ways to Handle Missing Values in Machine Learning | by Satyam Kumar | Towards Data Science

### Labelling

There are many types of modelling tasks that can be helpful in the policing environment. Some may be clearly defined like document labelling, others might be predictive, like risk assessment or future crime locations, others may be subjective like document summarisation. There will be different labelling best practices for different tasks, but in each case, to help prediction it is important to be very clear about what the goal of the modelling is and to define the task in a way that is best supported by the data.

- Make sure that the process that leads to the labels is clearly defined. For example, if you are replicating a human task, try to make sure borderline cases are well defined so there is consistency in labelling. Consistent labelling can improve your modelling more than any algorithm choice.
- If you are deciding labels based on data, consider all factors that influence the labels and clearly define the meaning of the label. Try to avoid shifting goalposts based on population statistics, e.g. 'top 5%' most violent criminals up to date, because this can change depending on the number of people you include in your sample group. Instead look for the patterns in that group and try to align them with theory and policy and potentially the input data features to ensure clarity.
    - If you are working on predicting risk of future harm. What constitutes harm, what are the outcomes that you are trying to prevent? Were there interventions that might have prevented the person from committing harm, such as prison terms, or protection orders? How do you model this knowledge?
      For example, can you record these as features if they are part of future events? Or do you disregard cases where someone's future risk was altered due to interventions? Test out best strategies.
    - If you are predicting event locations based on past events, are there interventions such as police presence or general unrest that have altered the labels in some way?
- In cases where you are predicting future events you may want to consider that your labels represent a noisy ground truth and use modelling techniques that account for noisy labels. https://arxiv.org/pdf/2108.04063v1.pdf
- The Ultimate Guide to Data Labeling: How to Label Data for ML | by SuperAnnotate | Geek Culture | Medium

### Assess and mitigate bias

Bias can be introduced at any point in the life cycle of data, e.g., how it was gathered, how it was entered, what is and is not available for modelling, word choice in text data, the locations that were sampled, societal impacts, etc. It is important to consider and record all possible sources of bias in the data cards.

[17] BEST: a decision tree algorithm that handles missing values | Computational Statistics

## Splitting Data

How you split the data will depend on the amount you have. The rule of thumb is to keep as much data as possible for training while allowing for a representative sample of different cases for testing, something like 80% for training, 10% for validation, and 10% for test. Key aspects of data splitting include:

- Making sure the different splits don't have overlapping examples. For instance, if you are using a suspect's case history, make sure there is no overlap between people in the different splits e.g., different crimes by the same suspect.
- Preserving the distribution of the test data. The test data should mirror the distribution of future values as much as possible. One way to do that is to randomly sample the data, preferably using stratified sampling to ensure enough of each class is in the test data. Alternatively, if the data is temporal, you can preserve a certain time span for test data.
    - While you may want to change the distribution of the training data, for example by under or oversampling a minority class, you should not change the distribution of the test data, although you can also look at subsets of test data to examine trends in errors, e.g. bias in a protected characteristic.
- As mentioned in the labelling section, errors can occur whether you are using manual or automatic labelling. Manually verifying the labels in the test data may improve accuracy. For example, if you have cases labelled low risk but the individual appears to be high risk, were there interventions in place that lead to that label?
- Validation data should be used to evaluate models in training and to choose a best model, the test data should be held out for final testing, otherwise the models will be overfitted and the test data will not give correct representation of future model accuracy.
- Cross-validation can be useful if there isn't much training data, but you should still try to have a held-out test set if possible or plan to collect test data during development so that it is ready for when you have likely models to compare.

Train Test Split in Deep Learning | by Igor Susmelj | Towards Data Science

## Modelling

### Knowing the data

Modelling is as much of an art as it is a science. It's important to familiarise yourself with the training data set and its properties to get the feeling for the likely algorithms that would work best for your data. #13 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 5]

- For tabular data:
    - One way to do this is to use a dataset visualisation package like pandas-dq or ydata-profiling, which are one of several choices people could make including deep learning-oriented ones like tensorflow data validation.
    - Clustering can also help uncover the landscape of the data, you can play around with different normalisation and preprocessing strategies to see if the data clusters become more coherent. You can compare the ground truth labels to see how well they align with clusters.
    - Visualising the labelled data also helps. For example, if the labels when plotted into 2D or 3D space (using a dimensionality reduction technique such as PCA) do not exhibit any clustering, it may indicate that you need an algorithm that can work with complicated and intermingled data. So, you might consider deeper neural networks or a combination of

multiple algorithms, all working on the same data or divided by modality (e.g. using a different type of classifier for text components and tabular components).

- Visualisation and exploration techniques also exist for timeseries, geospatial, and text data. For example:
    - For geospatial data:
        - Exploratory Spatial Data Analysis - an overview | ScienceDirect Topics
        - Analyze Geospatial Data in Python: GeoPandas and Shapely – LearnDataSci
    - Examples of text data exploration:
        - Although most advanced NLP modelling techniques look at full sentences and words in context rather than the n-gram and keyword exploration, it is useful to see what sort of patterns occur in your data A Complete Exploratory Data Analysis and Visualization for Text Data | by Susan Li | Towards Data Science
- Contrastive analysis can be a useful tool to how your dataset differs from general data, e.g. how does data specific to one crime differ from the data on all crimes mixed together. This works well with text for example comparing text in police reports against text in Wikipedia or newspaper corpora to see which phrases are outliers and how the language use is different might help you figure out what are salient features, or why some out of the box language modelling isn't working for you.
    - corpus-toolkit · PyPI
    - abidlabs/contrastive: Contrastive PCA
    - Corpus Analysis with spaCy | Programming Historian

## Knowing the data sources and biases

Knowing your data also needs to include understanding how the data was produced, and the procedures that could lead to idiosyncratic entries and variations in distribution over time. For example:

- The definition of a particular crime can vary over time and therefore lead to a loosely coherent set of features. This needs to be understood and recorded correctly in the data and model cards.
- Data entry systems have changed over the years, if using historical data, be mindful of shifts and consider mitigating strategies and feature engineering that could help override differences.
- Biases in the data can come from socio-economic, procedural, and other sources. Think about how the data you have has been collected and what could be the sources there. For example, whether it came from called-in crimes, or stop-and-search could make a big difference.
- If you are using locations, or if your data is predominantly from a particular community, you need to avoid a feedback loop where your model does not look outside the sample area.
- If you are using language modelling tools, consider examining the potential biases in the model that you are introducing and the fine-tuned model that you are producing.
- Fairness: Types of Bias | Machine Learning | Google for Developers

Where it is not practical to remove bias, quantify and minimise it where possible. The team should decide what is the appropriate quantitative and qualitative definition of discrimination to determine what is acceptable error, for example, what is the allowable difference in error rate for different groups.

Then test as much as possible if your model is amplifying these biases. There are several key steps that can be performed and documented:

- Test correlations between data features and protected attributes or sources of bias (e.g., ethnicity, location). Data features should be associated with the outcome rather than any protected features.
- For each level of each protected attribute the proportion in the overall population should be computed to learn which group errors may be more susceptible to estimation errors.

- If the correlations or group proportions are above a certain threshold (set by the team's acceptable error standards) minimise bias though:
    - Before Modelling: Relabelling, reweighting or resampling examples near the classification margin e.g. Mitigating Bias in Machine Learning: An introduction to MLFairnessPipeline | by Mark Bentivegna | Towards Data Science
        - Any training data adjustment strategies should be tested to ensure that the model performance is still correct.
        - Test data proportions should not be adjusted in any way.
    - During Modelling: using 'fairness regularisers' that consider differences in how the algorithm classifies protected vs. non-protected classes and penalises the model based on the extent of the difference. This is an active research area and it's worth looking for the right regularisers for your algorithm.
    - After Modelling: investigate bias with the test dataset

Fairness: Evaluating for Bias | Machine Learning | Google for Developers

Fairness | Machine Learning | Google for Developers

What does it mean to be fair? Measuring and understanding fairness | by Divya Gopinath | Towards Data Science

A Tutorial on Fairness in Machine Learning | by Ziyuan Zhong | Towards Data Science

## Understanding the model

It is incredibly easy to produce models by just putting data through a set of algorithms and picking some that perform best on some select criteria; however, pattern recognition algorithms can be very sneaky, and strive to find the easiest route to optimal performance. It is, therefore, important to understand if the resulting models are:

- leveraging all the information given or relying lazily on particular aspects of the data.
- biassed in a way that can affect a particular subset of the test data, whether this is applied to people, locations, incidents.
- just not addressing particular tough, rare and very important cases, and are instead relying on easy wins on more numerous instances.

By analysing the model performance on the development dataset you can learn about how the model is performing and learn how patterns in your data affect the performance.

- #14 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 6]
- #15 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 7]
- #16 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 8]

You should establish parameters for realistic performance, for example, by checking how well people do at your task:

- Andrew-NG-Notes/andrewng-p-3-structuring-ml-projects.md at master

- #12 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 2, Lesson 4]
- #30 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 3, Lesson 6]
- #31 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 3, Lesson 7]
- #38 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 3, Lesson 14]

Using ablation techniques, or using interpretability packages, model visualisation, and probative testing, as well as manual inspection of the errors can all help. Some examples include:

- About explainability: Picking an explainability technique | by Divya Gopinath | Towards Data Science
- Attention layer visualisation: Explainable AI: Visualizing Attention in Transformers - Comet
- Explainability tools: Building Trust in your ML Models — Explainability | by Matt Maufe | Filament-Syfter | Medium
  Explainable AI, LIME & SHAP for Model Interpretability | Unlocking AI's Decision-Making | DataCamp
- Explainability in time series: [2104.00950] Explainable Artificial Intelligence (XAI) on TimeSeries Data: A Survey
- In geospatial: The challenges of integrating explainable artificial intelligence into GeoAI - Xing - 2023 - Transactions in GIS - Wiley Online Library

It is important to understand the behaviour of your model with respect to protected characteristics of gender, race, and socioeconomic status. These must be recorded, the sources and reasons analysed, corrected if possible, or justified if not. If the algorithm is to be used in conjunction with human decision making these issues need to be clearly delineated and should inform how the model is deployed.

## General Links

- ML in Production course from Coursera is a good start for a refresher course on different aspects of developing a machine learning model that will be used as part of a larger system.[18] Other courses from deeplearning.ai are also very helpful.
- AI for good course from Coursera raises interesting topics of integrating AI in human-centric projects.[19]
- README.md - ashishpatel26/Andrew-NG-Notes · GitHub
- Landscape summary of algorithmic bias in AI
- Computational costs:
  - Profile model memory and CPU usage (v1) - Azure Machine Learning
  - https://aws.amazon.com/blogs/machine-learning/identify-bottlenecks-improve-resource-utilization-and-reduce-ml-training-costs-with-the-new-profiling-feature-in-amazon-sagemaker-debugger/
  - Pricing of LLMs Fine Tuning: now available with Azure OpenAI Service - Microsoft Community Hub

[18] #1 Machine Learning Engineering for Production (MLOps) Specialization [Course 1, Week 1, Lesson 1]

[19] #1 AI for Good Specialization [Course 1, Week 1, Lesson 1]

    - Reducing deep learning costs through model size reduction:
        - https://googlelambda.com/ai-faq/what-is-quantization-in-deep-learning
        - Quantization
        - artidoro/qlora - Efficient Finetuning of Quantized LLMs
        - AdapterHub
        - Methods and tools for efficient training on a single GPU
- Basics:
    - Introduction to ML and AI - MFML Part 1
    - Classification, regression, and prediction — what's the difference? | by Cassie Kozyrkov | Towards Data Science
    - What Is Your Model Hiding? A Tutorial on Evaluating ML Models
    - Crash Course in Data: Imputation techniques for Categorical features | by Akhilesh Dongre | AI Skunks | Medium
    - Machine Learning Crash Course | Google Developers
- Blogs and papers:
    - Papers With Code
    - NLP-progress
    - The Batch | DeepLearning.AI | AI News & Insights
    - Top ML Papers of the Week | LinkedIn
    - ruder.io
    - NeurIPS 2023
    - ICML 2024
    - ACL Anthology
    - Computation and Society
    - FATE: Fairness, Accountability, Transparency & Ethics in AI - Microsoft Research
    - https://research.google/pubs/?&category=responsible-ai

- Ethics and explainability:
    - Explainability:
        - About: Picking an explainability technique | by Divya Gopinath | Towards Data Science
        - SHAP and LIME examples for tabular data explainability
          Building Trust in your ML Models — Explainability | by Matt Maufe | Filament-Syfter | Medium
        - Geospatial: The challenges of integrating explainable artificial intelligence into GeoAI - Xing - 2023 - Transactions in GIS - Wiley Online Library
        - Explainability in time series: [2104.00950] Explainable Artificial Intelligence (XAI) on TimeSeries Data: A Survey
        - Explainable AI, LIME & SHAP for Model Interpretability | Unlocking AI's Decision-Making | DataCamp
        - Attention layer visualisation in transformers: Explainable AI: Visualizing Attention in Transformers - Comet
    - Fairness:

        - Fairness | Machine Learning | Google for Developers
        - What does it mean to be fair? Measuring and understanding fairness | by Divya Gopinath | Towards Data Science
        - A Tutorial on Fairness in Machine Learning | by Ziyuan Zhong | Towards Data Science
        - ML Fairness pipeline: Mitigating Bias in Machine Learning: An introduction to MLFairnessPipeline | by Mark Bentivegna | Towards Data Science
        - aif360 0.5.0 documentation
        - Fairlearn
    - AI and humans making decisions together: Responsible AI – getting the human back into the loop - Nokia Bell Labs
    - Computation and Society
    - Chat GPT and US law: https://peertube.dair-institute.org/w/o6sb7f7RwapWBJd9VC61t4
- Evaluations:
    - Evaluating in production vs academia: Evaluation Gaps in Machine Learning Practice
    - ML Flow example: How to Evaluate Models Using MLflow - The Databricks Blog
    - Evaluating in production without ground truth labels Introduction to NannyML Model Evaluation without labels | Pier Paolo Ippolito
    - A Comprehensive Guide on How to Monitor Your Models in Production
    - Adatest GitHub - microsoft/adatest: Find and fix bugs in natural language machine learning models using adaptive testing. and checkList GitHub - marcotcr/checklist: Beyond Accuracy: Behavioral Testing of NLP models with CheckList

- More technical videos:
    - Responsible AI – getting the human back into the loop - Nokia Bell Labs
    - Deep understanding of LLMs: Let's build GPT: from scratch, in code, spelled out.

- Books:
    - Deep Learning
    - A First Course in Machine Learning - second edition
- Seminar series:
    - Responsible AI seminar series - Nokia Bell Labs
    - VideoLectures
    - Ethics in AI Lunchtime Research Seminars
- Research groups:
    - Responsible AI – Google Research
    - FATE: Fairness, Accountability, Transparency & Ethics in AI - Microsoft Research
    - Safe and ethical | The Alan Turing Institute
- Useful code:
    - AutoGluon: AutoML for Image, Text, Time Series, and Tabular Data
    - AutoTrain – Hugging Face
    - auto-sklearn — AutoSklearn 0.15.0 documentation
- Bias:

- Debiasing data: An efficient framework using the principle of maximum entropy
- Debiasing Pretrained Text Encoders by Paying Attention to Paying Attention
- Bias benchmarking: https://arxiv.org/pdf/2110.08193.pdf
- https://www.microsoft.com/en-us/research/people/alexandrac/publications/

## Further Notes on Implementation for Data Scientists

It is entirely possible that a data scientist finds themselves at the point of deployment with a successfully trained model and nothing else in place. If an institution has good data collections that are easily leveraged, a data scientist can build a successful prototype without addressing many of the best practices suggested in the Development cycle, or indeed the Rationale. It is, therefore, imperative to stop before deployment and take stock of the ethical and engineering aspects of the project. Rationale, Unification, and Deployment sections should be filled out, and the model and data cards populated with all information relevant to the reproducibility and ethics of the project.

If proper tests and evaluations have not been conducted, if there is no baseline performance to judge the model against[20], if concerns of the community, minorities, or other affected parties have not been evaluated, discussed, and documented, this project will run into challenges.

## System and Model Integrity

In an environment where there can be frequent changes of personnel, a proper production system needs to be set up so that it is easily maintainable by any competent set of engineers and data scientists if it is to be deployed and used successfully on a regular and enduring basis. The models will require regular evaluation and retraining, so the project should contain:

- Automation: CI/CD pipelines
- Code quality and documentation: Ensure that you have well documented code that has been run through linters and code checkers.
- Unit testing: Any functions that transform data or influence the model in any way should be subject to unit tests. You can also write adversarial unit tests for the ML model with expected outcomes to ensure that retraining does not result in degradation of performance in specific types of cases, e.g. GitHub - marcotcr/checklist: Beyond Accuracy: Behavioral Testing of NLP models with CheckList . You can adapt the unit testing framework so that you pass a percentage of tests, as you may not be able to attain full performance on all cases you can design, right away.
- Data validation and quality: There should be format and quality checkers on all the incoming data to ensure that changes in inputs do not lead to prediction errors. There are packages out there than can help with this, e.g. Pandera
- Monitoring: Set up visualisations of the performance over time so that any degradation of performance is immediately visible, especially if you are using dynamic training. Ensure you track all relevant metrics to do with protected characteristics or potential feedback loop situations, as you want to be the first to spot any issues.
- Domain adaptation: Keep note of any interventions or changes in policy that may alter the distribution of data or have effect on the model performance.

Further information:

[20] the best baseline would be the current procedure (shadowing is a way to gather baseline data on how current procedure performs against the model)

- What's your ML test score? A rubric for ML production systems
- Production ML Systems | Machine Learning | Google for Developers
- Machine Learning Engineering for Production (MLOps)

## Integration into the Workflow

When it comes to integration of an ML model into the current procedures, it is important that there is a clear understanding of the limitations of a ML model. Machine learning can be helpful in policing in many different ways, but whether you have made a summarisation model, or a risk prediction one, one should not rely on the unchecked product of a model. A summarisation model might omit key information or potentially rephrase something in a way that has different implications. A model that predicts future actions that may or may not happen is an indicator of probability of such events at best, and as such interventions based on that should be cautiously applied and backed up with investigative insights.

- It is important that anyone directly interacting with model predictions is aware of the model limitations and any biases encoded within.
    - Regular training and reminders should be provided to users.
    - Limit the access to trained personnel only.
- Keep track of the procedural interventions, e.g. placement of police officers, protection orders, imprisonment, affect the expected outcome of the model. Integrate them into the future iterations of the model to avoid inaccuracy and obsolescence.
- Think about how the model is integrated into the workflow
    - Can you provide any contextualisation for the model decisions, e.g. examples similar to the one that is being considered or any salient features the model is considering
    - Provide warning about model limitations.
    - Provide a feedback button so that any errors or frustrations can be reported right away.
- Ensure that the way the model is used and triggered is complementary to how the model was trained.
    - For example, if the model is supposed to run each time a particular crime is reported, make sure that the same trigger was used to create training instances.
- Test how users are interacting with the model and how their reliance on, and trust in, the model varies over time.
- Ensure the deployment environment is safe and secure and that data is encrypted whenever possible, encrypted at rest or in transfer. [21]

## Data Scientist Expertise

Data scientist expertise needed at each stage are listed below.

### Rationale

- Experience with data modelling, ideally using police data
- Knowledge of explicit and implicit biases in the data e.g. how the data was compiled and possible issues inherent to the data
- Knowledge of processes involved in merging data from different sources, what variables need to be in the dataset given the model aims, what data is feasible given the different data systems

[21] https://thecyphere.com/blog/gdpr-encryption/

At this stage, knowledge of the potential data is key, and expertise in machine learning (ML) techniques less so

### Unification (data scientist and/or data engineer):

- Knowledge of what data is needed for the modelling and what the required final dataset should look like
- Knowledge of the current database systems and what can be retrieved for a) the modelling stage, and b) regular scheduled runs of the model in production
- Ability to engineer reliable pipelines for data extraction and unit testing to ensure that any changes in data or data inconsistencies are caught

### Development

- Ability to prepare data for ML e.g., which algorithms need scaled data, how to separate data into training/validation/testing sets etc.
- ML experience: sklearn at the minimum, KNN, forests/trees etc.
- Preferably some neural net experience, RNN, CNN
- Experience of natural language processing if leveraging text reports
- Ability to test models and troubleshooting issues
- Ability to assess model performance under various circumstances etc.

Knowledge of all algorithms is not required, but ML experience is essential. Some of the above can be learnt on the job if the person is given time, space and guidance.

### Implementation

- Ability to run the model in production (this requires good programming skills)
- Ability to test and maintain the model once it's in production
- Ability to perform unit testing and statistical testing to monitor performance in real time