# Machine learning at scale

Article • 02/11/2022 • 3 minutes to read • 8 contributors

In this article



Machine learning (ML) is a technique used to train predictive models based on mathematical algorithms. Machine learning analyzes the relationships between data fields to predict unknown values.

Creating and deploying a machine learning model is an iterative process:

- Data scientists explore the source data to determine relationships between *features* and predicted *labels*.
- The data scientists train and validate models based on appropriate algorithms to find the optimal model for prediction.
- The optimal model is deployed into production, as a web service or some other encapsulated function.
- As new data is collected, the model is periodically retrained to improve its effectiveness.

Machine learning at scale addresses two different scalability concerns. The first is training a model against large data sets that require the scale-out capabilities of a cluster to train. The second centers on operationalizing the learned model so it can scale to meet the demands of the applications that consume it. Typically this is accomplished by deploying the predictive capabilities as a web service that can then be scaled out.

Machine learning at scale has the benefit that it can produce powerful, predictive capabilities because better models typically result from more data. Once a model is trained, it can be deployed as a stateless, highly performant scale-out web service.

# Model preparation and training

During the model preparation and training phase, data scientists explore the data interactively using languages like Python and R to:

- Extract samples from high volume data stores.
- Find and treat outliers, duplicates, and missing values to clean the data.
- Determine correlations and relationships in the data through statistical analysis and visualization.
- Generate new calculated features that improve the predictiveness of statistical relationships.
- Train ML models based on predictive algorithms.
- Validate trained models using data that was withheld during training.

To support this interactive analysis and modeling phase, the data platform must enable data scientists to explore data using a variety of tools. Additionally, the training of a complex machine learning model can require a lot of intensive processing of high volumes of data, so sufficient resources for scaling out the model training is essential.

# Model deployment and consumption

When a model is ready to be deployed, it can be encapsulated as a web service and deployed in the cloud, to an edge device, or within an enterprise ML execution environment. This deployment process is referred to as operationalization.

# Challenges

Machine learning at scale produces a few challenges:

- You typically need a lot of data to train a model, especially for deep learning models.
- You need to prepare these big data sets before you can even begin training your model.
- The model training phase must access the big data stores. It's common to perform the model training using the same big data cluster, such as Spark, that is used for data preparation.
- For scenarios such as deep learning, not only will you need a cluster that can provide you scale-out on CPUs, but your cluster will need to consist of GPU-enabled nodes.

# Machine learning at scale in Azure

Before deciding which ML services to use in training and operationalization, consider whether you need to train a model at all, or if a prebuilt model can meet your requirements. In many cases, using a prebuilt model is just a matter of calling a web service or using an ML library to load an existing model. Some options include:

- Use the web services provided by Azure Cognitive Services.
- Use the pretrained neural network models provided by the Cognitive Toolkit.
- Embed the serialized models provided by Core ML for an iOS app.

If a prebuilt model does not fit your data or your scenario, options in Azure include Azure Machine Learning, HDInsight with Spark MLlib and MMLSpark, Azure Databricks, Cognitive Toolkit, and SQL Machine Learning Services. If you decide to use a custom model, you must design a pipeline that includes model training and operationalization.

For a list of technology choices for ML in Azure, see:

- Choosing a cognitive services technology
- Choosing a machine learning technology
- Choosing a natural language processing technology

# Next steps

The following reference architectures show machine learning scenarios in Azure:

- Batch scoring on Azure for deep learning models
- Real-time scoring of Python Scikit-Learn and Deep Learning Models on Azure

---

# Recommended content

### Many models ML with Azure Machine Learning - Azure Example Scenarios

Many machine learning (ML) problems are too complex for a single ML model to solve. Learn about many models machine learning at scale with Azure Machine Learning.

### Batch scoring of Python models on Azure - Azure Architecture Center

Build a scalable solution for batch scoring models on a schedule in parallel using Azure Machine Learning.

### Microsoft machine learning products - Azure Architecture Center

Compare options for building, deploying, and managing your machine learning models. Decide which Microsoft products to choose for your solution.

### Many models machine learning with Spark - Azure Example Scenarios

Many machine learning (ML) problems are too complex for a single ML model to solve. Learn about many models machine learning at scale in Azure with Spark.

### Azure Machine Learning decision guide for optimal tool selection - Azure Architecture Center

Learn how to choose the best services for building an end-to-end machine learning pipeline from experimentation to deployment.

### Machine learning DevOps guide - Cloud Adoption Framework

This guide provides a balanced view across three areas of MLOps. It discusses best practices and learnings from adopting MLOps in the Enterprise with Azure Machine Learning.

### MLOps for Python with Azure Machine Learning - Azure Reference Architectures

Implement a continuous integration (CI), continuous delivery (CD), and retraining pipeline for an AI application using Azure DevOps and Azure Machine Learning.

Show more

executive series

# The MACHINE LEARNING Primer

a SAS Best Practices e-book

by Kimberly Nevala

sas best practices

THOUGHT PROVOKING BUSINESS

# Table of Contents

1

# Machine Learning Defined

machine \mə-ˈshēn\ a mechanically, electrically, or electronically operated device for performing a task.

learning \ˈlərniNG\ the activity or process of gaining knowledge or skill by studying, practicing, being taught, or experiencing something.

# Do Machines *Learn?*

Yes! Machines learn by studying data to detect patterns or by applying known rules to:

- **Categorize** or catalog like people or things
- **Predict** likely outcomes or actions based on identified patterns
- **Identify** hitherto unknown patterns and relationships
- **Detect** anomalous or unexpected behaviors

The processes machines use to learn are known as algorithms. Different algorithms learn in different ways.  As new data regarding observed responses or changes to the environment are provided to the "machine" the algorithm's performance improves. Thereby resulting in increasing "intelligence" over time.

## But... Are Machines *Creative?* Or *Independently* Intelligent?

With the advent of big data, both the amount of data available and our ability to process it has increased exponentially. The ability of machines to learn and thus appear ever more intelligent has increased proportionally. Even so, machines aren't independent thinkers (yet).

Yes, machine learning may identify previously unidentified opportunities or problems to be solved. But the machine is not autonomously creative. The machine will not spontaneously develop new hypotheses from facts (data) not in evidence. Nor can the machine determine a new way to respond to emerging stimuli.

Remember: the output of a machine learning algorithm is entirely dependent on the data it is exposed to. Change the data, change the result.

CASE IN POINT

# Personalized Marketing

Companies are better than ever at understanding why customers buy their products, use their services, or engage their expertise. We can point the "machine" at a lake of consumer data to detect patterns and preferred channels for consumption. It can use historical and real-time data to determine that I, a frequent business traveler and coffee addict, may welcome a real-time message that my favorite coffee shop is around the corner. My dad would not welcome this interaction. He brews his coffee at home and will respond to a coupon in the mail. Which can also include incentives for other items he might buy on his next grocery outing.

The machine is optimizing activities for each customer across known channels (digital, paper, brick and mortar). It won't, however, independently create a new interaction channel that doesn't already exist.

# Problems That Lend Themselves to Machine Learning

In simple terms, machine learning is particularly suited to problems where:

- Applicable associations or rules might be intuited, but are not easily codified or described by simple logical rules.
- Potential outputs or actions are defined but which action to take is dependent on diverse conditions which cannot be predicted or uniquely identified before an event happens.
- Accuracy is more important than interpretation or interpretability.
- The data is problematic for traditional analytic techniques. Specifically, wide data (data sets with a large number of data points or attributes in every record compared to the number of records) and highly correlated data (data with similar or closely related values) can present problems for traditional analytic methods.

CASE IN POINT

# Identifying People and Things In Pictures

A practiced machine learning algorithm could recognize the face of a known "person of interest" in a crowded airport scene, thereby preventing the person from boarding a flight—or worse.

Social media platforms utilize machine learning to automatically tag people and identify common objects such as landmarks in uploaded photos.

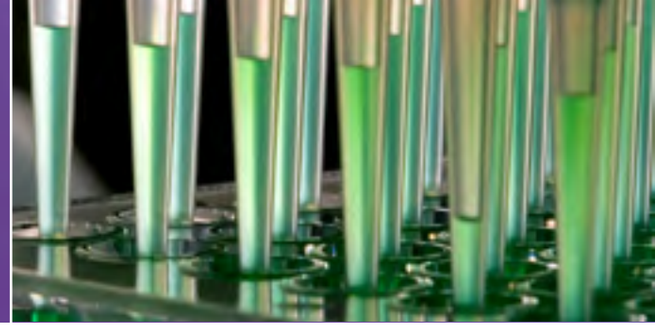
## Why Is This a Machine Learning Problem?

Image data is complicated. The number of pixels in each image make the data set wider than it is deep. Pixels close to one another have similar values making the data highly correlated. Images of the same subject have multiple subtle (and not-so-subtle) variations.

Of course, you can easily recognize people known to you - and those that aren't – in pictures; even when they have different expressions, poses or clothes. You can also identify "like" items both conceptually (i.e., animal, mineral or vegetable) and concretely (i.e., dog, cat, fish). But can you translate that knowledge into simple steps and discrete rules for how you made the match?

CASE IN POINT

# Genomics



Machine learning can help discover what genes are involved in specific disease pathways.

Machine learning can also be used to determine which treatments will be most effective for an individual patient based on their genetic makeup, demographic and psychographic characteristics.

## Why Is This a Machine Learning Problem?

Genomic data is wide: every person has more than 20,000 genes. As a result, the number of genes (data points) in an individual record is always larger than the number of people (records) in any data set.

A number of factors add to the complexity. Including, but not limited to: the high degree of variation within each of those 20,000+ genes. The fact that your relatives have similar genomes (making them highly correlated). That relatively few individuals may suffer from a given disease making the data pool extremely shallow. Last but not least, genes in isolation may not predict health outcomes or disease expression. Biochemical, environmental and other factors must also be considered, thereby requiring integrated data from multiple, diverse sources.

CASE IN POINT

# Navigation and the Self-Driving Car

Machine learning can identify the best routes from point A to B, predict transit conditions and travel time and predict the best route based on current, evolving road conditions.

Machine Learning can drive a car without requiring input from a driver.

## Why Is This a Machine Learning Problem?

Driving is a complicated but well-bounded problem. There are, in fact, a limited number of actions a vehicle may take: start, stop, go forward, go backward, turn, speed up and slow down. However, the decision to take any of action is influenced by numerous factors including but not limited to road conditions, weather conditions, presence and behavior of other vehicles, two-legged persons and their four-legged friends, and the rules of the road – just to name a few. While a human driver instinctually assesses all these inputs on the fly, capturing discrete rules for every possible combination is impossible.

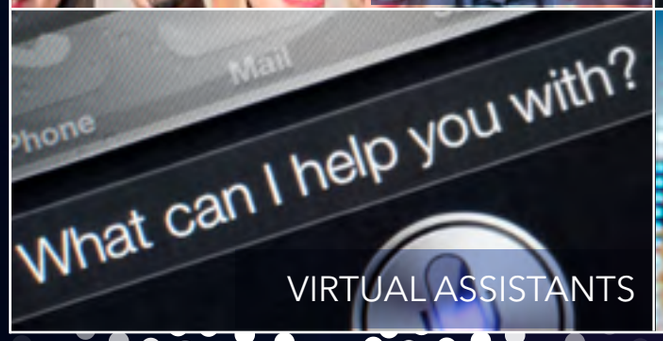
# Common Applications



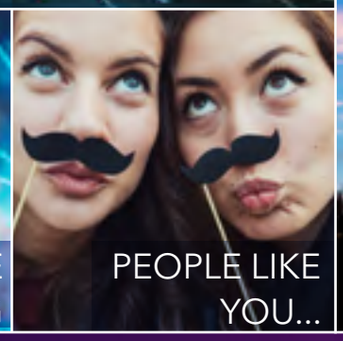












# 2

# The Basic Techniques

# The 4 Types of Machine Learning




# Supervised Learning

In supervised learning the machine is taught by example. Examples of the desired inputs and outputs are provided. The "machine" (aka the algorithm) uses this input to determine correlations and logic that can be used to predict the answer.

This is like giving students an answer key and asking them to "show their work." In supervised learning, sample Q&A are provided. The machine fills in how to get from A to B. Once the logical pattern is identified, it can be applied to solve similar problems.

## Common Techniques

- Bayesian Statistics
- Decision Trees
- Forecasting
- Neural Networks
- Random Forests
- Regression Analysis
- Support Vector Machines [SVM]

## Practical Applications






# Semi-Supervised Learning

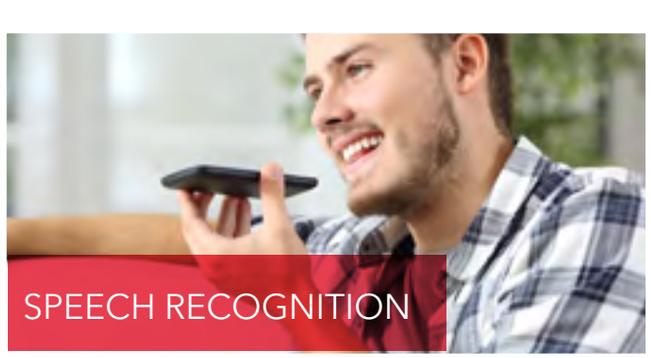
## Common Techniques

- See *Supervised Learning*

Semi-supervised learning is used to address similar problems as supervised learning. However, in semi-supervised learning the machine is provided some data with the answer defined (aka labeled) along with additional data that is not labeled with the answer. In other words, the some of the input data is tagged with desired output (answer) while the remainder is untagged.

Semi-supervised learning is used in cases where there is too much data or subtle variations in the data to be able to provide a comprehensive set of examples. In this case, the provided inputs and outputs provide the general pattern the machine can extrapolate and apply to the remaining data.

## Practical Applications


# Unsupervised Learning

## Common Techniques

- Affinity Analysis
- Clustering
- Clustering: K-Means
- Nearest-Neighbor Mapping
- Self-Organizing Maps
- Singular Value Decomposition

In unsupervised learning, the machine studies data to identify patterns. In this case, there is no answer key. The machine determines correlations and relationships by parsing the available data.

Unsupervised learning is modeled on how we humans naturally observe the world: drawing inferences and grouping like things based on unconstrained observation and intuition. As our experience grows (or in the case of the machine – the amount of data it is exposed to grows) our intuition and observations change and/or become more refined.

## Practical Applications



# Reinforcement Learning

## Common Techniques

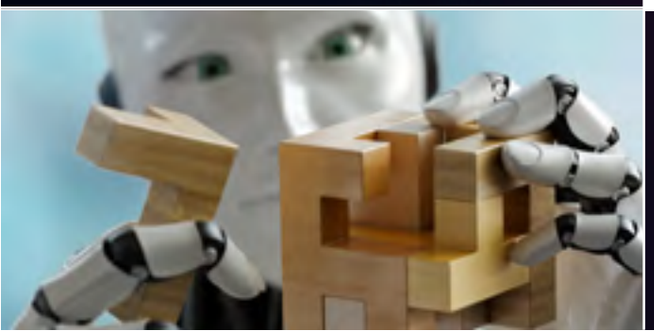
- Artificial Neural Networks (ANN)
- Learning Automata
- Markov Decision Process (MDP)
- Q-Learning

In reinforcement learning the machine is provided a set of allowed actions, rules and potential end states. In other words, the rules of the game are defined. By applying the rules, exploring different actions and observing resulting reactions the machine learns to exploit the rules to create a desired outcome.  Thus determining what series of actions, in what circumstances, will lead to an optimal or optimized result.

Reinforcement learning is the equivalent of teaching someone to play a game. The rules and objectives are clearly defined. However, the outcome of any single game depends on the judgment of the player who must adjust his approach in response to the incumbent environment, skill and actions of a given opponent.

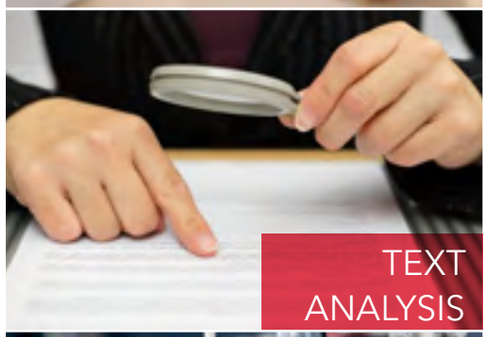
## Practical Applications



HOT TOPICS

# Deep Learning



A modern, advanced machine learning technique that makes use of extremely sophisticated neural networks. Called deep learning because the models generated are significantly more complex or deep than traditional neural networks. Deep learning models also ingest vastly larger amounts of data than their predecessors.

## Why Is This Important?

Deep learning is the underpinning of many advanced machine learning systems today. Perhaps most importantly, deep learning has vastly improved our ability to understand and analyze image, sound and video. This has been made possible by major advances in machine learning research as well as vast increases in both available data and massive computing power.

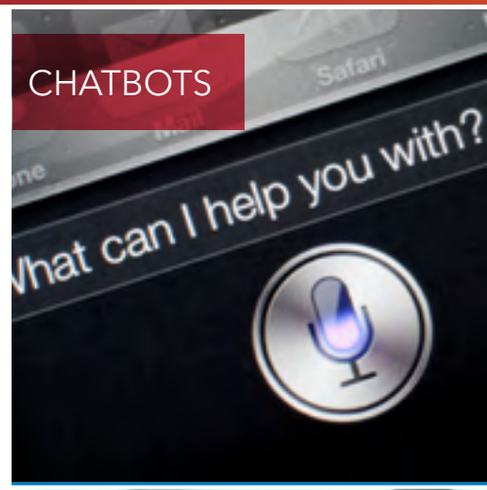
HOT TOPICS

# Cognitive Computing

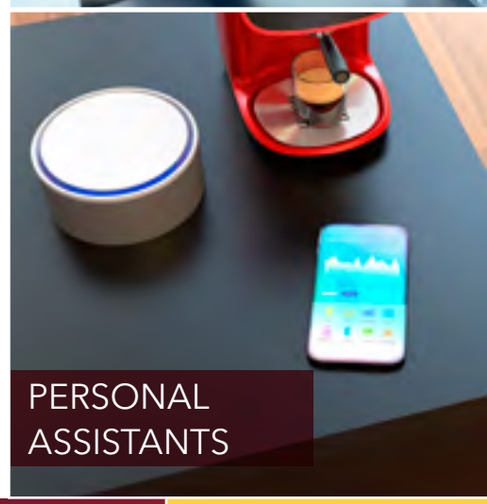
Systems that seek to understand and emulate human behavior. As well as to provide a more natural and intuitive interface between man and machine. This typically involves deploying systems that interface with people in their "native tongue." In other words, without requiring a user to write or understand code. Cognitive computing platforms accomplish this using a myriad of techniques including natural language processing, advanced machine learning algorithms (including deep learning) and natural language generation.

## Why Is This Important?

Cognitive computing makes machines (software systems) more accessible and intuitive to engage with. As a result, cognitive computing may be the key to increasing adoption of automated systems and analytic solutions. Ultimately, transcending the man vs. machine barrier in favor of cooperative systems in which man *and* machine seamlessly work together.  This is a precursor of what people commonly think of when speaking of artificial intelligence.







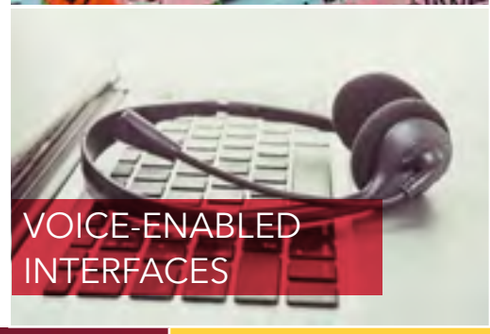

# Natural Language Processing

Capabilities that allow machines to understand written language, voice commands or both. Natural language processing (NLP) includes the ability to translate language into a form that a machine or algorithm can understand. Natural language generation (NLG) allows the machine to then communicate results or responses in “plain English” (or any language it’s designed to support).

## Why Is This Important?

Some NLP tools simply perform translation, mapping the words in a command to a dictionary. More sophisticated applications strive for understanding: inferring meaning or intent in order to inform an appropriate action or response. Given broad variances in dialects, figures of speech, colloquialisms, individual mannerisms and the rapid evolution of new modes of communication (abbreviations, emoticons) this undertaking is not trivial.






3

# Points to Ponder

# Why Can't Anyone Explain How the Machine Reached this Conclusion?

Unlike traditional statistical models, the models created by machine learning algorithms are extremely complex. While there is a method to the madness, it is not immediately obvious or linear. The exact path through a neural network, for example, is not easy to trace. Thousands (and even billions!) of rules or parameters can define the model. As a result, the exact internal processing pathways are a black box, even to the data scientist!

The more important question: is the algorithm or method being applied appropriately to the problem at hand?

# If **ML** Algorithms Are Black Boxes, How Can I Trust Them?

If the analytic mechanisms or - more specifically, the logical processing pathway or rules - are not clear or easily reproducible, how do you validate results?

Don't confuse black box processing with blind faith. When it comes to machine learning validation is deceptively simple.

When tested against new data:

- Does the algorithm accurately predict future events or result in desired outcomes?
- Can you put the output into action?

That's it. No more, no less.

# Is It Really *That* Simple?

The validation criteria for a machine learning algorithm are simple. The process of selecting, auditing and tuning an algorithm to deliver these results is anything but.

Numerous factors must be accounted for: what algorithm(s) best suit the problem or data? What data elements (aka features) should be included? Can the data be cleansed, transformed or refined to better expose key elements to the model (aka feature extraction and engineering)? How should the algorithm's parameters be configured or tuned for optimal performance?

The model must also be cross-validated and audited to avoid artificially engineering a deceptively accurate response (aka overfitting). An (admittedly) simplistic example: it is relatively easy to create a model that predicts yesterday's weather with a high degree of accuracy. But that same model may not predict tomorrow's weather as ably. Or be the only model that works.

Ultimately, developing a functional machine learning system is an iterative and intensive process that is part art and a lot of science.

# Is *Complicated and Clean* Always Better?

Not always. Like traditional analytics, a majority of time on machine learning projects is spent munging, validating and formatting data. But while data quality is always a concern, good enough is in the eye of the algorithm. Sometimes a simple algorithm with more data can often beat a complicated algorithm with less data. Even when the bigger data set is slightly dirtier.

When it comes to model accuracy, the higher the better. Or so it would seem; especially to the inexperienced. However, for many practical applications minute improvements in model accuracy will not result in germane operational improvements for the business. More data and features may also unnecessarily complicate the algorithm. The balancing act is between complexity and the ability to consume.

*Note! Andrew Ng, Chief Data Scientist for Baidu and a leading ML researcher, has posited that future advances in machine learning will be less about new algorithms and more about enabling algorithms to become smarter vis-à-vis the data that is fed into them.*

# Does **ML** Make My Existing Analytics *Obsolete?*

Machine learning is a tool in the analytics toolbox. Like any tool it must be thoughtfully applied lest it becomes the proverbial hammer looking for a nail. As machine learning emerged from academia, early adopters often found themselves expending significant time and effort on problems that could have been easily solved utilizing traditional statistical algorithms.

Therefore, machine learning is best seen as a supplement, not a wholesale replacement, for traditional analytic methods.

# Does **ML** Render *Humans* Obsolete?

Machine learning in practice requires human application of the scientific method and human communication skills.

The recipe is not as simple as: add data and stir.

Humans, above and beyond the data scientist programming the algorithm, are required to answer questions such as:

- What are we trying to predict?
- Are resulting correlations predictive? Causal? Are there inherent biases?
- Are results in line with expectations? Are there exceptions to be addressed?
- What is the predictive value and can it be generalized?
- Can the model and results be applied in real life?
- What is the proper response?

# 4

# Best Practices

Machine learning is a synergistic exercise between man and machine.

Machine learning in practice requires human application of the scientific method and human communication skills. Successful organizations have the analytic infrastructure, expertise and close collaboration between analytics and business subject matter experts.

# #1 Educate the Business on *Concepts*, not Theorems

Depending on your point of view, the inner workings of a neural network or deep learning algorithm are riveting. Or, horribly complicated and mind-numbing. The truth is that most people don't need (and aren't going to) understand the details.  Which isn't to say education isn't required. Making the case for time and funding requires executives and business layper-sons to broadly understand what machine learning can do.

A story that demonstrates how machine learning can be applied to your business will garner more engagement than complicated algorithmic charts and discussions of p-values . Rather than waxing rhapsodic about the technical nitty-gritty share examples of problems machine learning can solve. Include case studies from other industries and like companies.  If you must explain the method itself, think analogies not engineering diagrams.

# #2 Make Machine Learning Part of the Discovery Process

Machine learning algorithms observe behaviors or the environment, detect a pattern, make a generalization and infer an explanation or theory. The resulting probabilistic correlations may predict outcomes with a high degree of accuracy. They do not necessarily pinpoint the factors which create the outcome.

The bottom line? Prediction and causation are not the same. Business and policy decisions must consider this when deciding if and how to put found insights into action. In some cases, machine learning may identify areas for further study and consideration. In others, machine learning algorithms themselves might be integrated into operational systems to automate key decision points or processing pathways in real time.

## CASE IN POINT

Consider the following two cases:

The book *Freakonomics* highlights a case study in which the number of books in a home was correlated to high standardized test scores. The study led to a mayoral program to send free books home to poorly performing children. The results were decidedly less than stellar.

Contrast this with a study at the University of Ontario in which telemetry from devices attached to premature babies in neonatal intensive care was analyzed in real time. The systems predicted, with a high degree of accuracy, when a premature infant was developing an infection. Even though clinical symptoms did not present themselves until 48 hours later. The researchers and clinicians still do not know HOW the machine identified the onset of infection. But in this case, the important point was that it COULD. Ultimately, the team had to be comfortable working and acting on correlation, without fully understanding the causal relationship.

# #3 Avoid Black Box Exercises

Yes, machine learning methods can seem obscure and are, in fact, often inscrutable. But applying machine learning is not a black box activity. Humans, above and beyond the data scientist programming the algorithm, are required to answer questions such as:

### What are we trying to predict?

Like any analytic endeavor, machine learning projects should start with a clear statement of the problem space or hypothesis to be explored.

### What is best likely input into the process?

Data scientists and subject matter experts must work together to figure out the sources of data and the key features for the machine. Data visualization can play a key role in helping to highlight and test features that can be fed into machine learning algorithms.

*Note! Even in unsupervised learning the machine doesn't operate autonomously. Results are affected by decisions about what data to expose. The Google machine vision experiment independently identified pictures of cats. A different picture set would have resulted in a different entity being identified.*

# #3 Avoid Black Box Exercises *(cont'd)*

**Are results in line with expectations? Are there exceptions to be addressed? What are implications if they are not?**

Consider Stanford and Google's work in computer vision. While dang good, it's not foolproof. Goats get characterized as dogs, a field of tulips as hot air balloons. Minor gaffs to be sure, but what are the implications when people are incorrectly categorized?

**How can (and should) results be applied?**

Machine learning is great at determining what to do. Not necessarily so good at defining how (although this is changing fast).

**What is the proper response?**

For instance, when a pattern emerges with global health or political ramifications what is the proper next step or steps?

# #4 Apply Appropriate Scientific Rigor

Machine learning does not negate the need for solid statistical reasoning, scientific and data analysis. While extremely powerful, it is not a magical, self-correcting analytical panacea. At least not yet.

Machine learning is most effective when the data scientist has a solid understanding about how to structure the system. In other words, the developer can identify the appropriate algorithm(s) based on the domain and how attributes of the data will respond. The process requires both knowledge of the characteristics of different algorithms and a healthy dose of intuition.

*Forewarning! Even with experience, developing a machine learning application is an experimental and iterative process: regardless of whether the team is using well-known algorithms. In every case, the algorithms must be trained and tuned for the business context and data at hand.*

Teams must also apply a healthy (but not paralyzing) dose of skepticism and rigor to validate the model lest they fall into the trap of "believing everything they think."

## CASE IN POINT

Examples of failure to critically analyze analytic models abound.

In one instance, a team of genomics scientists created an algorithm for predicting a patient's response to chemotherapy. Unfortunately, they didn't account for data variations and data integrity issues in their initial training data set. This led to some unfortunate consequences, the least of which were canceled clinical trials.

In another, economists published a paper adversely linking GDP growth with high government debt. Questions were later raised regarding factor weighting used in the regression model which, when modified, led to dramatically different conclusions.

# #5 Make it *Just* Complicated Enough

Generally speaking, more data wins. But do more features (aka attributes or data points) also equate to better outcomes? Not always. The catch-22 is that larger data sets beget larger variations in data and a larger potential for predictive error.

In many cases, weaker models with more data do better than more complicated models with less.  Even if the data is dirtier. This is the basis of ensemble modeling techniques which use the 'power of the collective' to predict results.

Deploying machine learning artfully is a balancing act. One in which the incremental predictive value of complexity must be weighed against interpretability, ease of use and applicability.

Of course, simplicity is not the only virtue. Ultimately, the model must also perform well under realistic operating conditions.

## CASE IN POINT


The best model is for naught if it can't be put into operational practice. Consider the Netflix Prize in which Netflix challenged the data science community to create a highly performing model to predict "what you should watch next." The winning model exceeded all expectations; predicting user preferences at an unprecedented level.

The problem? The model's data and processing requirements made it impossible to execute in real time or near real time – a key requirement when trying to attract the attention of users looking for a flick to watch right now! The result? A fantastically accurate model with no applicable business value.

# #6 Actively Engage Business Users in Validation

The data scientist must apply due diligence as the model is created and tuned. To do so, they must effectively communicate and collaborate with data and business domain experts to validate and vet the model. This is critical to ensure the team has, amongst other things:

**Validated that all the options have been considered.**

A model being significantly accurate doesn't preclude other equally well-fitting models from existing. And those models may suggest alternate conclusions.

**Accounted for potential bias.**

Algorithms – for all that they feed on data – are not inherently fool-proof or unbiased. Beyond subconscious biases unwittingly introduced, the data provided to a model can reflect biases by virtue of the decisions by which the data was created. As mathematician Jeremy Kun elegantly stated, "training on human-generated data (aka found data) means the inherent biases of a population (minority or majority) or the underlying process will be inherited."

**Identified the impact and implications of putting found insight into action.**

# #7 Employ Data Storytelling

Like other techniques, machine learning insights can suffer a failure in translation between the data science teams and end consumers. Therefore, in addition to questions posed above, the team must carefully consider how found insights will be delivered and consumed.

To start, teams must determine how to present in a manner that is palatable and consumable. This is particularly important when found insights challenge existing paradigms or require changes to standard operating procedures. Rather than slaying your audience with numbers and statistical values, can the results be visualized and supported by a compelling story?

*Forewarned! Don't make up a compelling story. Rather, create a narrative that shows how the findings drive operational improvements or enable innovative new products and services – in terms and context the audience understands.*

# # 8 Don’t Underestimate the Power of Perception

With machine learning, an individual’s future actions can be predicted with a high degree of confidence. Creating the potential for what data scientist John Foreman, MailChimp’s chief data scientist and self-professed “recovering management consultant” called “laser-guided disingenuous arguments” for targeting and marketing. The ethics (including the privacy vs. value debate) are important and must be addressed upfront.

Another consideration: how does our expectation of fallibility change when a machine decides?

As “machines” increasingly encroach into visible decision-making roles and engagement we must account for the human reaction. Building trust in these systems is critical for full adoption and engagement.

## CASE IN POINT

What happens when a self-navigating car crashes? How likely are we to forgive that accident? Even if it's a mistake a human driver makes with greater frequency.

Studies have shown machine learning to be 72 percent accurate in diagnosing breast cancer from a mammogram. A human doctor? 65 percent. But does a patient's expectation change when a machine makes or misses the diagnosis?

# #9 Proactively Adapt Business Processes

Deploying machine learning systems may require us to be willing to give up control as we automate certain decisions and actions. Machine learning may also enable development of entirely new products, services or customer engagement models.

Careful consideration must be given to the resulting business implications.

- Is the organization ready and willing to take action and make necessary changes to incorporate found insights?
- What existing business processes and roles must be modified?
- Will net new processes or roles be required?
- If the machine learning model will be autonomous, how will automated systems work within the context of human workflows?  Make sense to their human counterparts or co-workers?

## CASE IN POINT

Manufacturers are applying machine learning to identify potential equipment failures just in time, before they happen. This requires fitting equipment with sensors and embedding analytic sensing systems. It also precipitates a fundamental rethink of customer service, maintenance and warranty policies and procedures.

Merchandisers are utilizing machine learning for real-time online pricing. In this case, the machine determines optimal price points while gross thresholds are validated by humans. A feedback loop allows the algorithm to learn from observed sales results. Another takes buyer input on missed opportunities or errors.  This seemingly simple shift required a seismic change in how merchandisers and buyers were measured and incented.

In Amazon's automated distribution centers humans do the packing while robotic systems collect required supplies and validate the right stuff gets in the box. What makes it all work? The interaction between human and robot. Underscoring the point that deliberate design of not just the algorithm, but the ongoing engagement between man and machine, is critical to success.

# #10 Plan for Ongoing Care and Feeding

To stay current and deliver results, machine learning algorithms must be continuously refreshed and refined based on data that reflect current circumstances. This is true whether evaluating the impact of new customer micro-segments on retention or rebalancing the network to account for unexpected spikes in energy demands to avoid blackouts.

If you apply machine learning successfully – incenting consumers to buy more, utilize different commercial channels, and turn right not left – the patterns used to create the model are changed. Machine learning models need to adjust to account for these shifts in behavior. Your business processes do too.

In this case, operationalization is not a merely keep the lights on activity. Maintaining the model is a critical, ongoing process that requires as much, if not more, due diligence as initial model development.

5

# Are You Ready?

CHECKLIST

# Are You Ready for Machine Learning?

## Articulate a Problem That Needs Solving

Machine learning works best when there is a clear problem statement. The problem definition should include actions to be enabled and/or measurable outcomes to be achieved. Even better? If the problem is clearly tied to top-of-mind operational challenges or strategic objectives.

Because machine learning is time and data intensive, a critical evaluation of whether existing analytic models/approaches or alternate solutions may apply is also in order. This ensures potential value is commensurate with input effort.

*Note! Routine decision points that are high-volume, require immediate or rapid response and/or are dependent on highly variable inputs are good candidates for machine learning.*

CHECKLIST

# Are You Ready for Machine Learning?

## Establish an Experimental Mindset

Machine learning is an iterative, experimental process. Although core algorithms are increasingly commoditized, every project must be customized based on the business context and data. As with any good experiment, some hypotheses will be proven false. New data may need to be procured or created. Or the problem statement recast based on what is found. As a result, decision makers and team members alike must adopt a test-and-learn mentality for machine learning to succeed. Use a gated, iterative process that provides the flexibility and agility to quickly assess progress to determine whether an alternate approach is warranted or when enough is enough.

## Enlist a Collaborative Data Science Team

ML expertise is a requirement. Equally important is a dynamic teaming model that engages diverse experts with business, data and technical expertise. This includes data experts that can assess and onboard requisite data assets, business experts to provide context, assess implications (business, social, moral) of proposed actions or new product or service offerings and IT personnel who deploy and maintain the technical ecosystems. Not to be overlooked: resources who can translate between the language of the "quant" (i.e. those who speak math) and that of the business.

CHECKLIST

# Are You Ready for Machine Learning?

## Develop a Robust Data Strategy and Ecosystem

Machine learning runs on data. A lot of data. Establishing a data process to effectively identify, acquire (or create), provision and access high-quality data and information assets is critical. To that end, governance policies and the data ecosystem must support exploratory environments (often referred as sandboxes) as well as production environments. This requires a multi-tiered approach to balance access and agility without sacrificing security, privacy, or quality. The introduction of non-traditional (big) data sources including unstructured text, voice, pictures and so on may also require new data management capabilities.

## Assess the Organization's Risk Tolerance

From agreeing on the criteria for what is "good enough" to understanding how to validate and develop models, machine learning often challenges traditional approaches to quality assurance and risk management. Why? At some point, the training wheels or, in this case, the training data must come off. Real validation comes from testing the performance of the machine against new data. Very often, this entails putting the system into operational practice. This may range from performing "A/B testing" in production to confirm a model will incent desired customer behaviors to taking a self-driving car on the road with a human overseer ready to take the wheel in a crisis.

CHECKLIST

# Are You Ready for Machine Learning?

## Commit to Adapting Established Business Processes

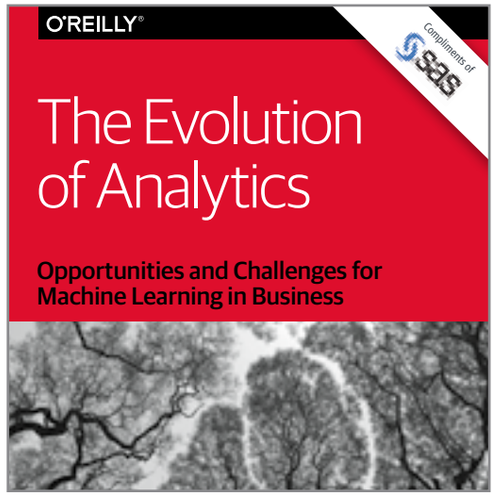
Whether automating an existing decision point or enabling a net new product or service offering, ML is disruptive. Assessing potential implications to existing business processes, functions and roles is key. This doesn't mean architecting the entire change before you start. But a quick gut check can mitigate the potential for costly moot exercises. To plan for the plan, begin by asking: "If we answer this question or provide this hypothesis, what can we do with the information? How might this impact existing processes? Are we willing and able to make the requisite changes?"

## Commit to Adopting New IT Practices

Once deployed, iterative modeling and tuning of the machine learning model should continue. The cadence at which updates are required is unpredictable, not conforming to traditional scheduled deployment windows. As a result, deploying machine learning requires fundamentally different QA and deployment models, skill sets and service levels than traditional IT DevOps practices.

RESOURCES

# Want to Learn More?




## *The Evolution of Analytics: Opportunities and Challenges for Machine Learning in Business*

Take a closer look at how modern machine learning applications are delivering business value today. Includes two case studies highlighting steps organizations are taking to utilize machine learning to discover the insight hidden inside their data.

## *An Executive's Guide to Cognitive Computing*

SAS EVP and CTO Oliver Schabenberger demystifies cognitive computing by sharing relatable examples and key lessons learned on where and how to apply this emerging capability for ultimate effect.

sas best practices

THOUGHT PROVOKING BUSINESS

# About the Author

**KIMBERLY NEVALA** is the Director of Business Strategies for SAS Best Practices. Kimberly brings 19 years of on-the-ground experience advising clients worldwide to help organizations maximize their data potential. She is responsible for market analysis, industry education, emerging best practices and strategies in the areas of business intelligence and analytics, data governance and management.

A speaker and author, Kimberly is often consulted on the topic of strategic enablement and organizational dynamics. Her work has been featured on Information Week, CIO Asia, Knowledge World and TDWI. Kimberly is the author of *The Anatomy of an Analytic Enterprise, Sustainable Data Governance* and *Top 10 Mistakes to Avoid When Launching a Data Governance Program.*





**SAS Institute Inc.**
100 SAS Campus Drive
Cary, NC 27513-2414, USA
Phone: 919-677-8000
Fax: 919-677-4444
Email: bestpractices@SAS.com

Published in Towards Data Science



Rahul Agarwal Follow

Sep 7, 2019 · 9 min read ★ · Listen



Pixabay



# Six Important Steps to Build a Machine Learning System

## A field guide to thinking about ML projects

Creating a great machine learning system is an art.

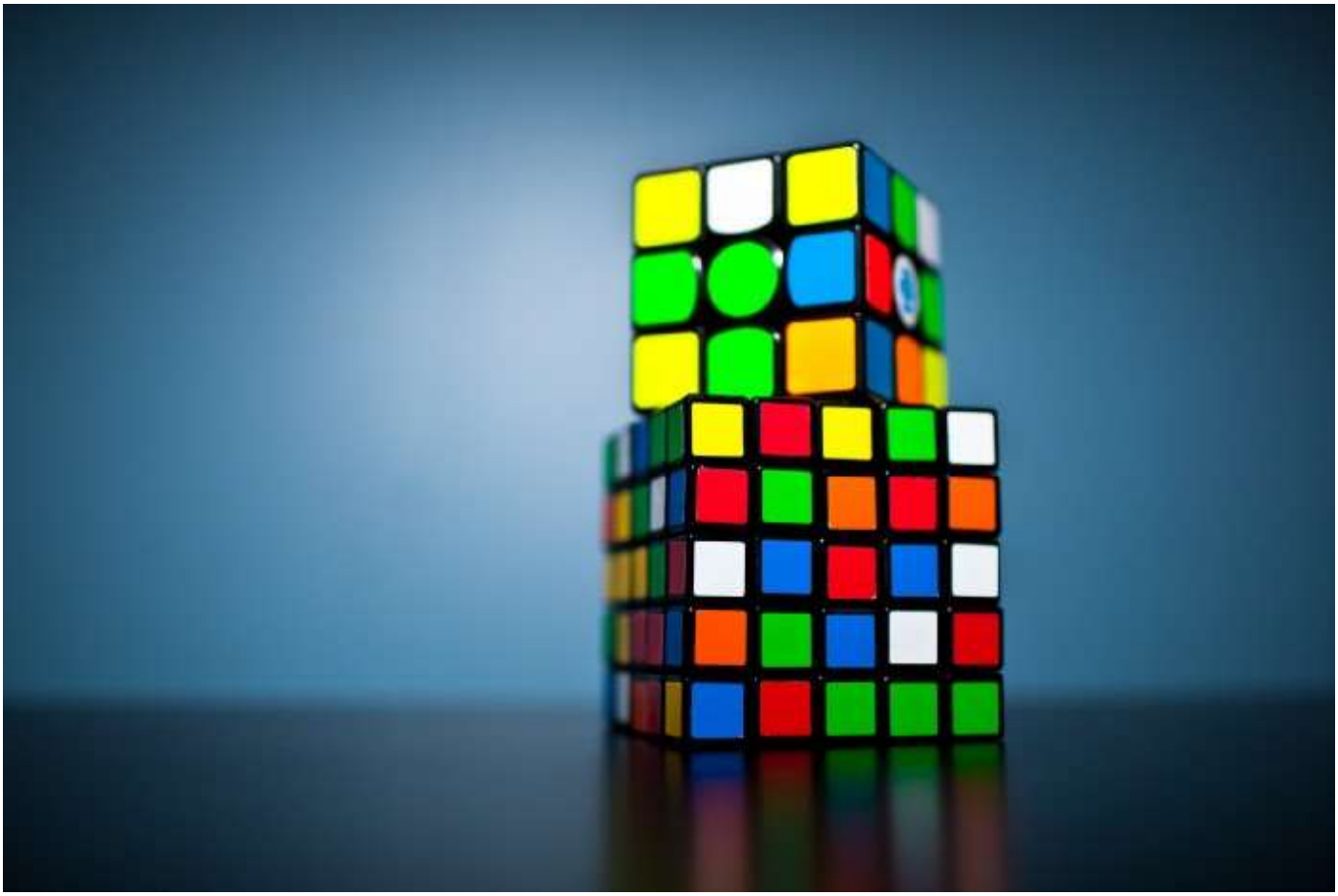
There are a lot of things to consider while building a great machine learning system. But often it happens that we as data scientists

# A machine learning pipeline is more than just creating Models

It is essential to understand what happens before training a model and after training the model and deploying it in production.

***This post is about explaining what is involved in an end to end data project pipeline. Something I did learn very late in my career.***

## 1. Problem Definition



This one is obvious — ***Define a problem.***

And, this may be the most crucial part of the whole exercise.

***So, how to define a problem for Machine learning?***

Well, that depends on a lot of factors. Amongst all the elements that we consider, the first one should be to understand ***how it will benefit the business***.

That is the holy grail of any data science project. If your project does not help business, it won't get deployed. Period.

Once you get an idea and you determine business compatibility, you need to ***define a success metric***.

***Now, what does success look like?***

Is it 90% accuracy or 95% accuracy or 99% accuracy.

Well, I may be happy with a 70% prediction accuracy since an average human won't surpass that accuracy ever and in the meantime,

For example: For a click prediction problem/Fraud application, a 1% accuracy increase will boost the business bottom line compared to a 1% accuracy increase in review sentiment prediction.

> Not all accuracy increases are created equal

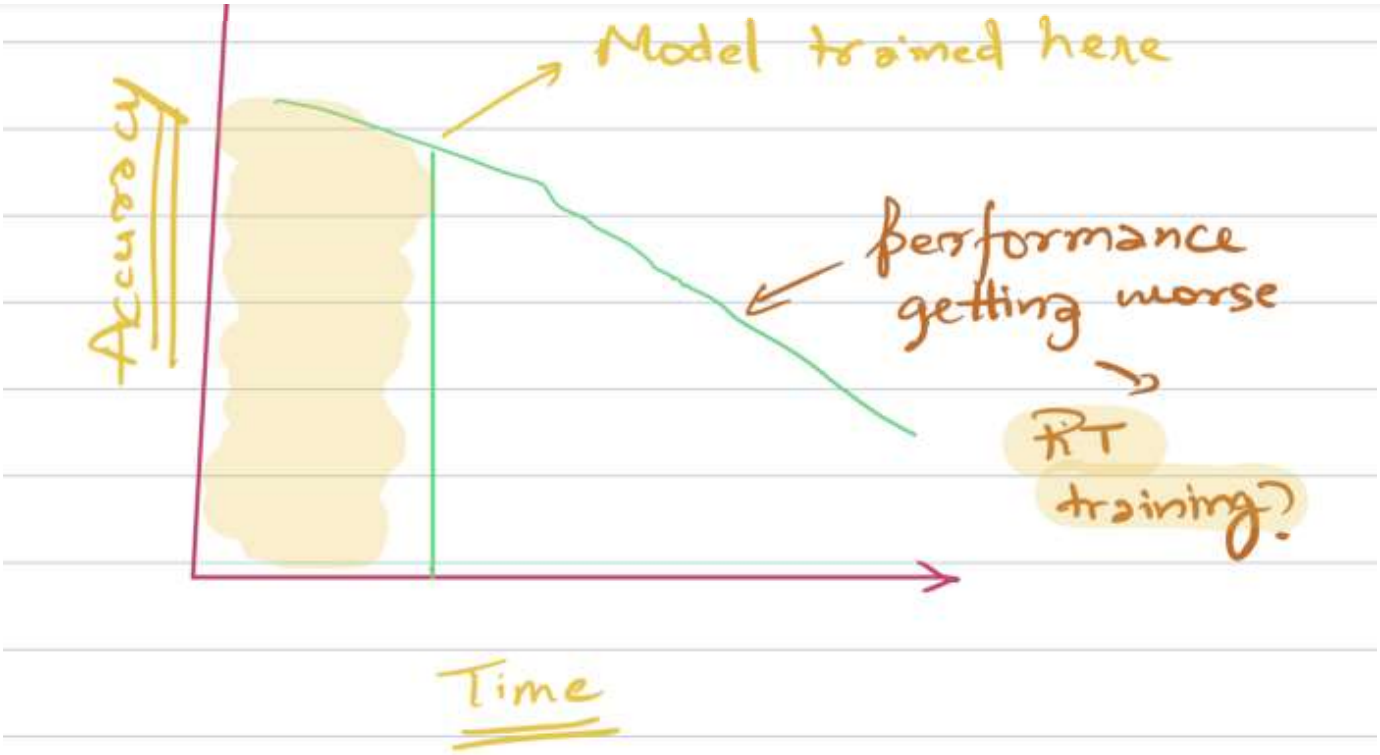
### 2. Data


There are several questions you will need to answer at the time of data acquisition and data creation for your machine learning model.

The most important question to answer here is: ***Does your model need to work in realtime?***

If that is the case, you can't use a system like Hive/Hadoop for data storage as such systems could introduce a lot of latency and are suitable for offline batch processing.

***Does your model need to be trained in Realtime?***




If the performance of your ML model decreases with time as in the above figure, you might want to consider Real-time training. RT training might be beneficial for most of the click prediction systems as internet trends change rather quickly.

***Is there an inconsistency between test and train data?***

Or in simple words — ***do you suspect that the production data comes from a different distribution from training data?***

For example: In a realtime training for a click prediction problem, you show the user the ad, and he doesn't click. Is it a failure example? Maybe the user clicks typically after 10 minutes. But you have already created the data and trained your model on that.

There are a lot of factors you should consider while preparing data for your models. You need to ask questions and think about the process end to end to be successful at this stage.

### 3. Evaluation

How will we evaluate the performance of our Model?

***The gold standard here is the train-test-validation split.***

Frequently making a train-validation-test set, by sampling, we forgot about an implicit assumption — Data is rarely ever IID(independently and identically distributed).

***In simple terms, our assumption that each data point is independent of each other and comes from the same distribution is faulty at best if not downright incorrect.***

For an internet company, a data point from 2007 is very different from a data point that comes in 2019. They don't come from the same distribution because of a lot of factors- internet speed being the foremost.

If you have a cat vs. dog prediction problem, you are pretty much good with Random sampling. But, in most of the machine learning models, the task is to predict the future.

You can think about splitting your data using the time variable rather than sampling randomly from the data. For example: for the click prediction problem you can have all your past data till last month as training data and data for last month as validation.

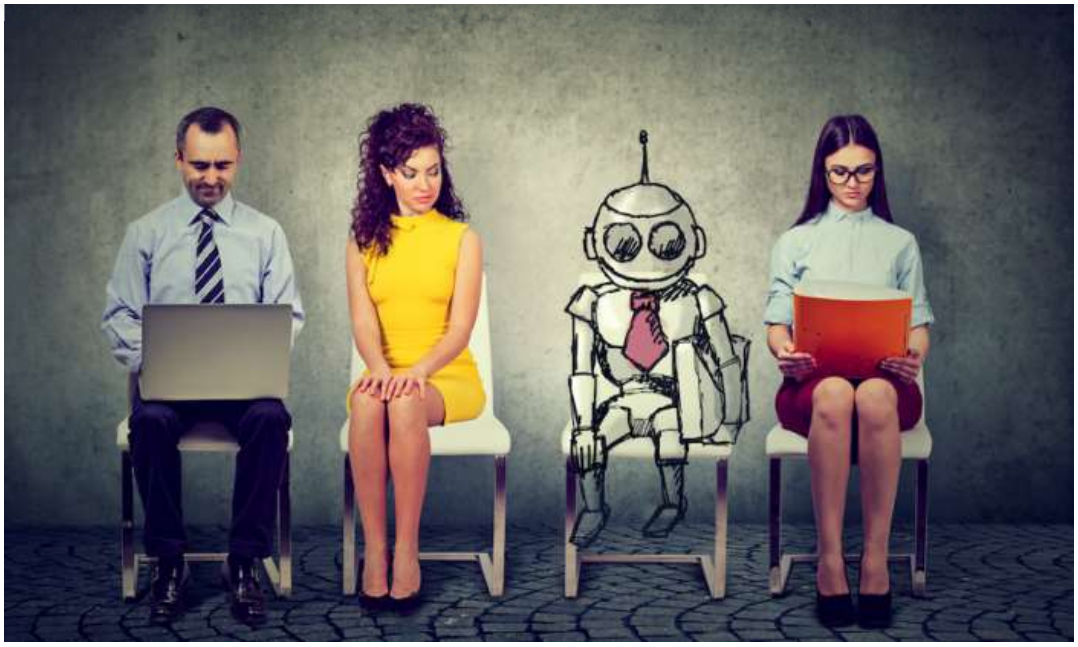
***The next thing you will need to think about is the baseline model.***

Let us say we use RMSE as an evaluation metric for our time series models. We evaluated the model on the test set, and the RMSE came out to be 4.8.

Is that a good RMSE? How do we know? We need a baseline RMSE. This could come from a currently employed model for the same task. Or by using some simple model. For Time series model, a baseline to defeat is last day prediction. i.e., predict the number on the previous day.

For NLP classification models, I usually set the baseline to be the evaluation metric(Accuracy, F1, log loss) of Logistic regression models on Countvectorizer(Bag of words).

***You should also think about how you will be breaking evaluation in multiple groups so that your model doesn't induce unnecessary biases.***



Last year, Amazon was in the news for a secret AI recruiting tool that showed bias against women. To save our Machine Learning model from such inconsistencies, we need to evaluate our model on different groups. Maybe our model is not so accurate for women as it is for men because there is far less number of women in training data.

Or maybe a model predicting if a product is going to be bought or not given a view works pretty well for a specific product category and not for other product categories.

Keeping such things in mind beforehand and thinking precisely about what could go wrong with a particular evaluation approach is something that could definitely help us in designing a good ML system.

### 4. Features

Good Features are the backbone of any machine learning model. And often the part where you would spend the most time. I have seen that this is the part which you can tune for maximum model performance.

> Good feature creation often needs domain knowledge, creativity, and lots of time.

On top of that, the feature creation exercise might change for different models. For example, feature creation is very different for Neural networks vs. XGboost

**The Hitchhiker's Guide to Feature Extraction**
An exhaustive look at feature engineering techniques
towardsdatascience.com

Once you create a lot of features, the next thing you might want to do is to remove redundant features. Here are some methods to do that

**The 5 Feature Selection Algorithms every Data Scientist should know**
Bonus: What makes a good footballer great?
towardsdatascience.com

## 5. Modeling


Interpretable ML Book

Now comes the part we mostly tend to care about. And why not? It is the piece that we end up delivering at the end of the project. And this is the part for which we have spent all those hours on data acquisition and cleaning, feature creation and whatnot.

So what do we need to think while creating a model?

The first question that you may need to ask ourselves is that ***if your model needs to be interpretable?***

There are quite a lot of use cases where the business may want an interpretable model. One such use case is when we want to do attribution modeling. Here we define the effect of various advertising streams(TV, radio, newspaper, etc.) on the revenue. In such cases, understanding the response from each advertisement stream becomes essential.

Apart from model selection, there should be other things on your mind too:

- ***Model Architecture:*** How many layers for NNs, or how many trees for GBT or how you need to create feature interactions for Linear models.
- ***How to tune hyperparameters?:*** You should try to automate this part. There are a lot of tools in the market for this. I tend to use hyperopt.

### 6. Experimentation

Now you have created your model.

It performs better than the baseline/your current model. How should we go forward?

We have two choices-

1. Go into an endless loop in improving our model further.
2. Test our model in production settings, get more insights about what could go wrong and then continue improving our model with ***continuous integration.***

I am a fan of the second approach. In his awesome third course named Structuring Machine learning projects in the Coursera Deep Learning Specialization, Andrew Ng says —

> *"Don't start off trying to design and build the perfect system. Instead, build and train a basic system quickly — perhaps in just a few days. Even if the basic system is far from the "best" system you can build, it is valuable to examine how the basic system functions: you will quickly find clues that show you the most promising directions in which to invest your time."*

You should always aim to minimize the time to first online experiment for your model. This not only generated value but also lets you understand the shortcomings of your model with realtime feedback which you can then work on.

### Conclusion

> Nothing is simple in Machine learning. And nothing should be assumed.

You should always remain critical of any decisions you have taken while building an ML pipeline.

A simple looking decision could be the difference between the success or failure of your machine learning project.

***So think wisely and think a lot.***

This post was part of increasing my understanding of the Machine Learning ecosystem and is inspired by a great set of videos by the Facebook engineering team.

If you want to learn more about how to structure a Machine Learning project and the best practices, I would like to call out his awesome third course named Structuring Machine learning projects in the Coursera Deep Learning Specialization. Do check it out.

Thanks for the read. I am going to be writing more beginner-friendly posts in the future too. Follow me up at **Medium** or Subscribe to my **blog** to be informed about them. As always, I welcome feedback and constructive criticism and can be reached on Twitter @mlwhiz.