10 Things That Can Go Wrong with ML Projects (and what you can do about it)



Google Cloud



Karl Weinmeister {Developer Advocacy Manager}

@kweinmeister

Machine learning practitioners are solving important problems every day. They're also experiencing a new set of challenges. Let's discuss the best practices and tools to help.

What can go wrong?

Building a Model

- 1. You aren't solving the right problem
- 2. Jumping right into development without a prototype
- 3. The training process is slowing your team down

Model accuracy

- 4. You have an imbalanced dataset
- 5. Model accuracy is not good enough

Transparency and Fairness

- 6. The model doesn't serve all of your users well
- 7. It's unclear how your model works

MLOps

- 8. You accidentally push a bad model into production
- 9. Your model accuracy is drifting downward
- 10. Model inference isn't scaling well in production

Building a Model

1. You aren't solving the right problem

Some organizations are **transforming** how they operate and serve their users with machine learning.

Others **struggle to get value** out of their machine learning projects.

What **goal** is your machine learning model trying to achieve?

How do you answer whether your model is "good" or "bad"? Do you know what your **baseline** is?



Focus on a long-term mission with maximum impact



Insights from the <u>Product Management for Al video</u>:

- Stay focused on your goals, but be **flexible on the tactics** to get there
- Set milestones, but be prepared to pivot
- Your **users** should be involved at each stage of the project

Ensure that your problem is a good fit for machine learning

Predictive Analytics

Unstructured data

Fraud detection Preventive maintenance Click-through-rate Demand forecasting Will customer buy? Annotate videos Identify eye disease Triage emails

Automation

Schedule maintenance Reject transactions Count retail footfall Scan medical forms Triage customer emails Act on user reviews

Personalization

Customer segmentation Customer targeting Product recommendation

2. Jumping into development without a prototype

A machine learning project is an **iterative process**.

You should start with a **simple model** and continue to **refine it** until you've reached your goal.

A quick prototype can **tell you a lot** about hidden requirements, implementation challenges, scope, etc.



Create models directly from BigQuery with BQML



Execute ML initiatives without moving data from BigQuery

Iterate on models in SQL in BigQuery to increase development speed

Automate common ML tasks

Built-in infra management, security & compliance, supporting a seamless transition to production model development

Automate the model building and deployment process with Cloud AutoML



3. Model training can take a long time

Model training is the process of learning from input data to construct a model.

For many real-world datasets, **training can take hours**, **days**, **or even longer**.

After your model is trained, you can then evaluate its accuracy to see if you've made an improvement.

When your team is trying to rapidly iterate on new ideas and techniques, **training time can slow down projects and get in the way of innovation**.



Serverless Training with Vertex AI

- Train models without managing infrastructure
- Supports TensorFlow 2.x and other popular data science and machine learning frameworks. You can even run your own Docker container.
- Leverage distributed training on the latest GPUs and TPUs to finish jobs faster
- Improve your model quality with automated hyperparameter tuning using parallel trials

48	AI Platform	← Job Details	DEPLOY MODEL DOWNLOAD MODEL			
51	Dashboard	caip_training_1	597268920			
1	Al Hub	Succeeded (4 min 18 s				
		Creation time	Aug 12, 2020, 4:48:44 PM			
64	Data Labeling	Start time	Aug 12, 2020, 4:49:58 PM			
Ē.	Notebooks	End time	Aug 12, 2020, 4:53:02 PM			
		Logs	View Logs			
44	Pipelines	TensorBoard	TensorBoard is available from this page only for mo			
12	Jobs	Consumed ML units	0.06			
:=	2003	Training input	V SHOW JSON			
9	Models	Training output	V SHOW JSON			
		Model location	gs://healthy-terrain-225716/			

Google Cloud TPU





\$

Cloud TPUs enable businesses and researchers to <u>train and run</u> cutting-edge & large-size ML models at faster speed and scale. With Cloud TPUs, you can train more powerful and accurate models than ever before. Cloud TPUs can reduce the cost of machine learning work and speed up time to market for new AI applications.

Model Accuracy

4. You have an imbalanced dataset

Some ML tasks, such as fraud detection, use data with many more labels of one class than another.

If 99% of data is benign, and 1% is fraudulent, it's easy to get 99% accuracy in your model. Just predict that every transaction is benign!

How can you balance accuracy across each class?



Techniques to deal with imbalanced data:

- Weighting each class
- Oversampling and undersampling
- Generating synthetic data



tensorflow.org/tutorials/structured_data/imbalanced_data

AutoML -Optimization Objective

When training a model, select an appropriate optimization objective for your project goals:

- AUC PR (maximize precision-recall curve for less common class)
- AUC ROC (distinguish between classes)
- Log loss (maximize accuracy across entire dataset)

Advanced opt	ions 🔨		
Optimization	objective		
	ne outcome you're trying to ach	eve, you may want to train	your model to
optimize for a d	ifferent objective. Learn more		
AUC ROC Distinguish t	between classes		
C Log loss Keep predict	ion probabilities as accurate as po	ossible	
AUC PR Maximize pre	ecision-recall curve for the less cc	mmon class	
Precision	At recall value	0	
🔿 Recall	At precision value	0	

AutoML -**Model Evaluation**

Review accuracy metrics 1. at different thresholds

.

- Review confusion matrix 2. to understand where misclassifications occur
- 3. Consider updating **Optimization Objective** and re-training if desired



Confusion matrix @

A confusion matrix helps you understand where misclassifications occur (which classes get "confused" with each other). Each row is a predicted class and each column is an observed class. The cells of the table indicate how often each classification prediction coincides with each observed class.



5. Model accuracy is not good enough

Sometimes, you might feel stuck.

You can't improve the model accuracy any more.

Is there anything that can be done?





Understanding the Reporting of Causes of Flight Delays and Cancellations

Solutions that may help

- Improve domain expertise for the problem you're trying to solve
- Include more, varied training data
- Feature engineering
- Consider removing features that may be causing overfitting; or start with a smaller model and incrementally add features
- Try different model architectures, hyper-parameter tuning, and ensembles
- Try AutoML to see what's possible with your input data

Transparency and Fairness

6. Your model doesn't serve all of your users well

"Fairness is the process of understanding bias introduced by your data, and ensuring your model provides equitable predictions across all demographic groups. Rather than thinking of fairness as a separate initiative, it's important to apply fairness analysis throughout your entire ML process."

- Sara Robinson, <u>Building ML models for everyone:</u> <u>understanding fairness in machine learning</u>



Responsible AI: Questions for each phase of the project

the mod	oblem(s) will del solve? he intended	data col sampleo	d, labeled? resentative of	How wa trained? Who tra When?		tested?	s the model st datasets ed?	Is the mode behaving as expected? Why did the fail in this ca	e model
Define p	What are the associated w use-case?			Train mo	How was the debugged/im	proved?	e	Deploy & monitor	
	What will 'sue look like?	ccess'	privacy-prote	ecting?	limitations?	models			

What-If Tool: Visually probe the behavior of trained ML models

Visualize	Binning X-Axis (none) -	Binning Y-Axis (none)	Color By Therence label 1 There	Label By (default) =	Scatter X-Axis	Scatter Y-Axis	:
Datapoints Partial dependence plots Show nearest counterfactual datapoint							
Show similarity to selected datapoint O							
Edit < > ① T II O, Storch features							
Select a datapoint to begin exploring model behavior for your selection. Edit and Infer: Edit your datapoint here and run Inference in the Infer Table to see differences in model behavior. Visualize: Switch between visualizing datapoints and exploring partial dependence plots to gain insights into your model's behavior. Explore counterfactuals or see how similar (or different) the rest of your dataset is from your selection.	+	99100	3502		0.994	Legend Calors	~
Infer ^						by Inference label 1 <=50k	1
Run inference	-					● >50k	

Google Cloud

TensorFlow Model Analysis: Sliced ML model metrics

In [13]: # Show data sliced along feature column trip_start_hour.
tfma.view.render_slicing_metrics(
 tfma result 1, slicing column='trip start hour')





ר קיזו ^ב קיזו ^ב קיזו ²	wip-wip-wip-wip-wip-wip-
---	--------------------------

feature	accuracy	accuracy_baseline	auc	auc_precision_recall	average_los
trip_start_hour:8	0.97938	0.97938	0.66513	0.99010	0.1111
trip_start_hour:9	0.98113	0.98113	0.69231	0.99140	0.0892
trip_start_hour:10	0.95197	0.95197	0.77377	0.98236	0.1541
trip_start_hour:1	0.94180	0.94180	0.78422	0.98231	0.1901

7. It's unclear how your model works

ML models are often distributed without a clear understanding of they function.

How will you be able to explain how your model makes predictions?

Under what conditions does the model perform best and most consistently?

Does it have blind spots? If so, where?



Explainable AI on Google Cloud: Support for multiple data types

Images



Text

How could you not love cake?!

Tabular	•
---------	---

Name	Feature value	Attribution value
distance	1395.51	-2.44478
start_hr	18	-1.29039
max_temp	20.7239	0.690506
temp	16.168	0.12629
dew_point	7.83396	0.0110318
prcp	0.03	-0.00134132

Sentiment score: 0.9



Document your Model with Model Cards



modelcards.withgoogle.com

Evolute

tensorflow / model-ca	rd-toolkit	⊙ Watch + 14	습 Star 102 및 Fork 17
Code 🕚 Issues 👘	Pull requests 3 💿 Actions 📃 Projects	🖽 Wiki 🗇 Secu	rity 🗠 Insights
P master - P 5 branche	os 🛇 0 tags Go to file Add file -	± Code +	About
	ess-infra-github no-op 📾 4538be3 yesterda	/ ③40 commits	a tool that leverages rich metadata and lineage information in MLMD to build
github	Internal change	3 months ago	a model card
model_card_toolkit	no-op	yesterday	D Readme
CODE_OF_CONDUCT.md	Internal change	3 months ago	Apache-2.0 License
CONTRIBUTING.md	Internal change	3 months ago	
D LICENSE	Internal change	3 months ago	Releases
README.md	Add model card image to README.md	3 months ago	No releases published
B RELEASE.md	publish Standalone MCT Case Study	13 days ago	
🗅 setup.py	Add Markdown template to ModelCardToolkit outp	16 days ago	Packages
			No packages published

README.md

Model Card Toolkit

The Model Card Toolkit (MCT) streamlines and automates generation of Model Cards [1], machine learning documents that provide context and transparency into a model's development and performance. Integrating the MCT into your ML pipeline enables the sharing model metadata and metrics with researchers, developers, reporters, and more.

Some use cases of model cards include:

Contributors 5

Q. .0

Languages

 Python 50.4% . HTML 49.6%

Model Card Toolkit

MLOps

8. You could accidentally push a bad model into production

Your data science team is constantly running experiments to find a better model.

What if you accidentally deploy a model that wasn't tested properly, and your users are impacted?



Create a reproducible workflow with Vertex Pipelines

- An ML pipeline allows you to run the same steps in the same environment every time with a trigger or schedule
- Include steps for:
 - a. Data validation
 - b. Model evaluation
 - c. Conditional deployment
- Track pipeline runs and generated artifacts



9. Your model accuracy is drifting downward

Over time, the performance of most models will decay.

Conditions in the outside world change. This means that the data distributions for the features the model were trained on will also change.

How do you detect this data drift and manage it?



Detect and manage model drift with MLOps processes

Continuous Evaluation

- Sample your model's predictions
- Compare the model's predictions to "ground truth"
- Assess the accuracy
- Send notification if threshold reached

Continuous Training

- Deploy an ML pipeline that automatically:
 - Extracts latest data from sources
 - Trains a model
 - $\circ \quad \ \ \text{Tests the model} \\$
 - Deploys to production

Vertex Model Monitoring

Automatically alert your data scientists and ML engineers when model performance changes

Detect drift and training-serving skew

Provides confidence in model reliability





10. Model inference isn't scaling well in production

Your ML model is a success.

It solves the problem well, and will be integrated into a widely used application.

How will you host the model and serve all of the requests?



Robust and reliable model hosting with Vertex Prediction

- Serve online endpoints for low-latency predictions, or predictions on massive batches of data
- Scale **automatically** based on your traffic
- Log prediction requests and responses to BigQuery for monitoring and debugging
- Choose from a variety of compute options, including the inference-optimized NVIDIA T4 GPU, for faster predictions

AI Platform	← Model deta	IS + NEW VERSION CREAT	TE ENDPOINT START BATCH	PREDICTION	SHOW INFO PANEL
Dashboard Al Hub 🔄 Data	Name continuous_aval_mo	del	Default version foo		
Notebooks	VERSIONS	EVALUATION			
Clusters	👻 Filter by prefix				0 C
Jobs	Name	Create time	Last used	Evaluation	Labele
I Models	🗋 💿 bar	May 4, 2018, 2:47:41 PM	Sep 12, 2018, 11 OR 12 AM	No plan	owner : jason
AI Platform	🗋 🔿 fds	May 4, 2018, 2,47:55 PM	Sep 12, 2018, 10:45:25 AM	Manual, 300/day	owner : jason
	Built	on			
		gle Kube	ernetes E	Ingine	(GKE)

Resources

Resources



Vertex Al

Try out the end-to-end platform for data science and machine learning on Google Cloud

<>> Codelabs Q. Search Google Codelabs Doogle Developers Codelate provide a guided, tutorial, hands on coding expension. Most codelates will step you through the process of building a small application, or adding a new feature to an existing application. They cover a wide range of topics such as Android Wear, Google Compute Engine, Project Tango, and Google APIs on ICG. A-Z Record Choose at event + Filter by category A Tour of Cloud to'T Core Accelerated Mobile Pages Advanced Accelerated Mobile Pages Foundations Concepts Take a tour of Google Cloud to? Core by creating a This codelab will introduce you to the key concepts of registry, adding a device to the registry, and This codelab will introduce you to the key concepts of Accelerated Mobile Pages and how they differ from connecting a virtual device before replacing the virtual Accelerated Mobile Pages and how they differ from traditional HTML documents. We will achieve this by device with a real one powered by Android Things. traditional id?Mt documents. We will arbitrat this for building and validating AMP documents. building and validation AMP documents. Access files in Cloud Storage with the Activity Recognition Transition API Add Ads Support to a CAF Receiver Spring Resource abstraction Codelab In this codelati, you'l build a Cast Receiver application that uses the new Ads features. Learn to access files in Cloud Storage with the Spring Learn how to use Activity Recognition Transition Api Resource abstraction to build powerful contextual features in your app

Codelabs

Hands-on labs that enable you to learn machine learning concepts and tools on Google Cloud



Al Adventures

50+ Youtube videos covering machine learning concepts and how to use Cloud Al Platform

Thank you!

Have questions or suggestions? We want to hear from you! Please contact <u>@kweinmeister</u>!