

# Deep Learning on Kubeflow

---

Iurii Mozzhorin, Rebekka Pech

July 15, 2019

Hands-on Lab - Big Data Technologies SS 2019

Goethe University Frankfurt am Main

# What is Kubeflow?

- an open-source platform
- based on Kubernetes
- simple, portable and scalable deployments of ML
- integration of existing tools and libraries

## **Includes components for**

- model developing, training, serving
- hyperparameter tuning
- pipelines



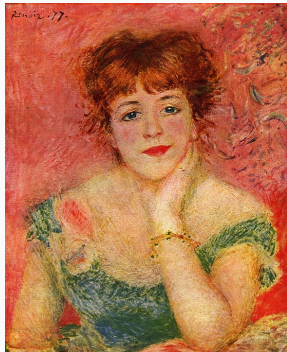
# Use case

## Goal

- Train a CNN model to distinguish between works of 10 painters

## Dataset

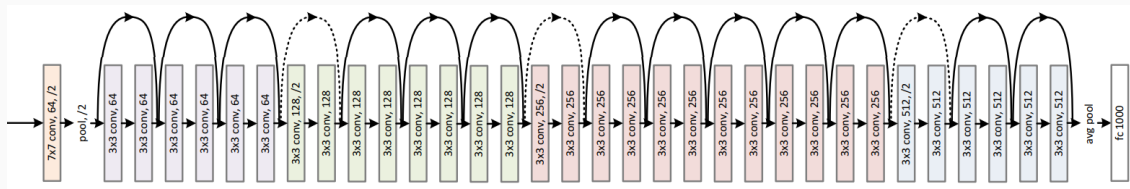
- "Painter by Numbers" from Kaggle.com
- Original size 83GB
- Images dimensions 477 – 7444 px
- Selected & resized:
  - 340 MB
  - 388 images  $\times$  10 artists
  - images smallest dimension 448 px
- Test set 20 images (5%)



# Use case

## Model

- ResNet 50
- Transfer learning with weights from ImageNet (1000 classes)
- Input size  $224 \times 224$  px
- Image augmentation: shift, rotation, flip



- Preprocess the data
- Create the model (transfer learning)
- Create the Docker image
- Train the model on KF using TFJobs on CPUs
- Save the trained model
- Train the model on KF using TFJobs on GPUs
- Distributed training on multiple pods
- Serve the model

- Create the model: TF → Keras
- Containers: Docker and Kubernetes
- Manage pods (TFJobs): kubectl + ksonnet → kubectl + kustomize → kubectl
- Manage KF deployment: GCP console, gcloud + gsutil
- Manage data: PVC → GCS bucket
- Monitoring: Tensorboard
- Manage the model: TF Serving → Jupyter Notebook

- **Keras** simplifies work with CNN models but have several restrictions:
  - cannot work directly with GCS bucket
  - does not (directly) support distributed learning
- **Tensorflow** requires additional learning time
- **Kubeflow** is very raw
  - rapid changes, low backward compatibility
  - requires learning of all underlying technologies
- $\Rightarrow$  **Kubeflow does not suit for research in data science**

## References

- <https://www.kubeflow.org/docs/>
- <https://github.com/kubeflow/examples>
- <https://www.kaggle.com/c/painter-by-numbers/>
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Deep Residual Learning for Image Recognition, arXiv:1512.03385
- <https://keras.io/>
- <https://www.tensorflow.org/guide>
- <https://kubernetes.io/docs/>



Thank you for your attention!