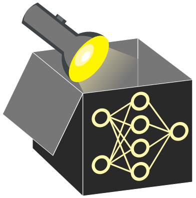
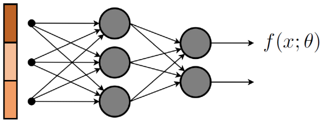
Interpretable Machine Learning

Simple Gradients & Integrated Gradients

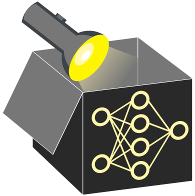




Learning goals

- Basics of sensitivity analysis
- Saliency maps for images and language
- integrated gradients

SENSITIVITY ANALYSIS

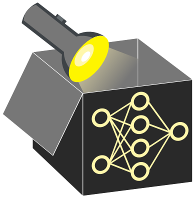


- Neural Networks are differentiable machines
  - The output can be written as a function of the parameters and input
  - One can differentiate the output function w.r.t parameters
  - The underlying idea is used for training Neural Nets using gradient descent

$$f(x;\theta) \qquad\qquad \frac{\partial f(x;\theta)}{\partial \theta}$$

- Sensitivity Analysis: How sensitive is the output $f()$ w.r.t to a small change in the input?

SENSITIVITY ANALYSIS



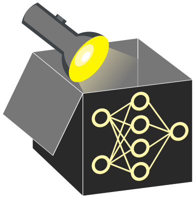
- Neural Networks are differentiable machines
  - The output can be written as a function of the parameters and input
  - One can differentiate the output function w.r.t parameters
  - The underlying idea is used for training Neural Nets using gradient descent

$$f(x;\theta) \qquad\qquad \frac{\partial f(x;\theta)}{\partial \theta}$$

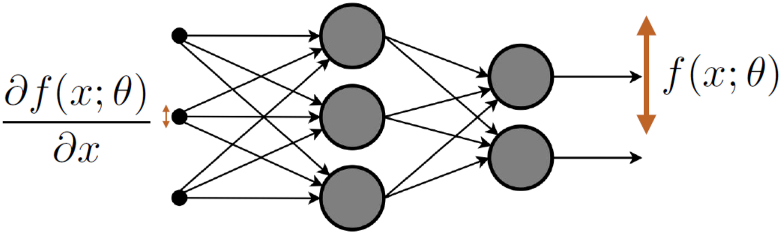
- Sensitivity Analysis: How sensitive is the output $f()$ w.r.t to a small change in the input?

$$\frac{\partial f(x;\theta)}{\partial x}$$

SENSITIVITY ANALYSIS

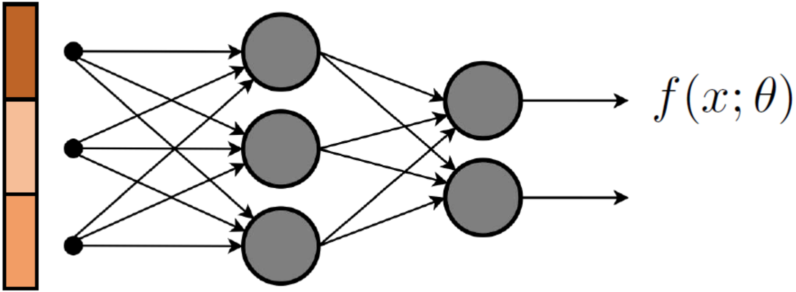


- How sensitive is the output $f()$w.r.t to a small change in the input ?
  - If a small change in the input feature causes a large change in output, then that feature is responsible for the prediction
  - Back-propagation into the input: instead of computing $\frac{\partial f(x;\theta)}{\partial \theta}$



SALIENCY MAPS

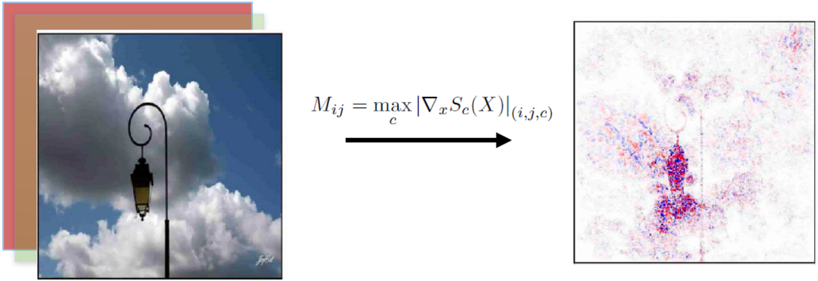
- Visualize the gradients over each feature
  - as a heat map or Saliency Maps
  - Saliency maps are feature attribution methods that are based on gradients



SALIENCY MAPS FOR IMAGES

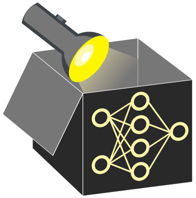
- Images have multiple channels where each channel is a 2-D matrix

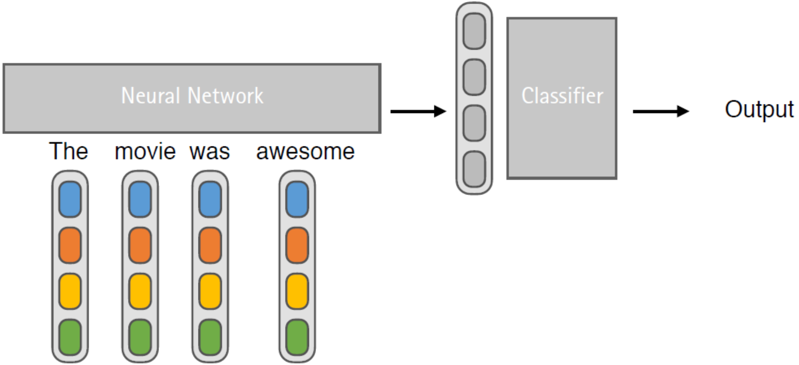
$$M_{ij} = \max_{c} |\nabla_x S_c(X)|_{(i,j,c)}$$



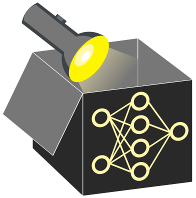
SALIENCY MAPS FOR LANGUAGE

- Words are associated with an embedding
- Computing gradients back to the inputs is different in comparison to images

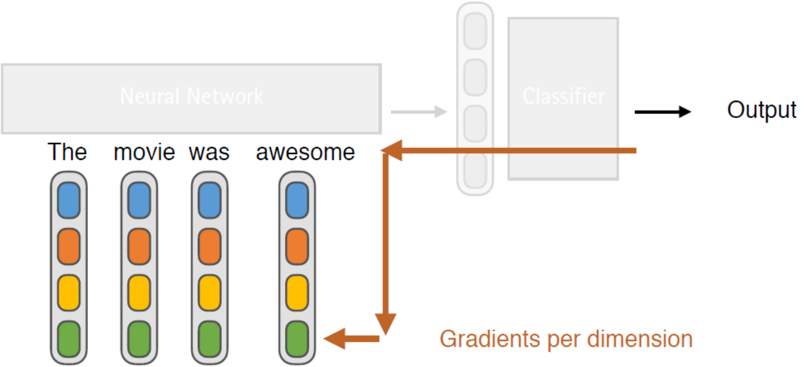




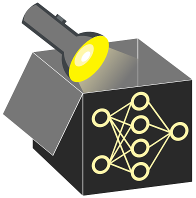
SALIENCY MAPS FOR LANGUAGE



- We obtain gradients per dimension but we want attributions or importance scores at the level of world
- **Idea:** Simple aggregations of dimension-level gradients like sum, average, etc.



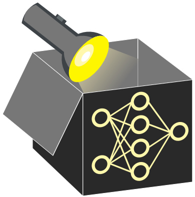
SALIENCY MAPS - SETTING



Which features are responsible for the decision given..

A trained model M	Post-hoc interpretability
An instance x	Local interpretability
Access to model parameters	White-box interpretability

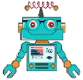
SALIENCY MAPS - SETTING



Which features are responsible for the decision given..

A trained model M	Post-hoc interpretability
An instance x	Local interpretability
Access to model parameters	White-box interpretability

Input



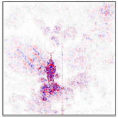
“Pole”

Output

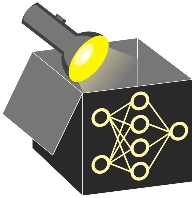
Saliency Maps

Heatmaps

Feature Attributions

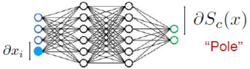


SALIENCY MAPS - SETTING

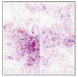


Which features are responsible for the decision given..

A trained model **S** | An instance **x** | Access to model parameters



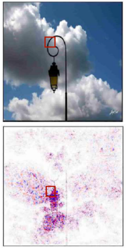
A feature is more relevant if a small perturbation causes large change in the output

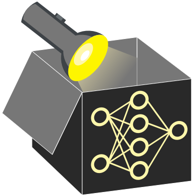


Saliency Map

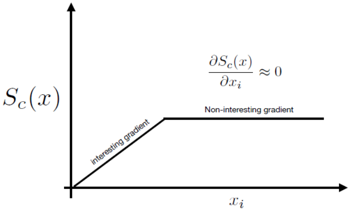
$$R_i^c(x) = \frac{\partial S_c(x)}{\partial x_i}$$

PROBLEMS WITH DEEP NETS





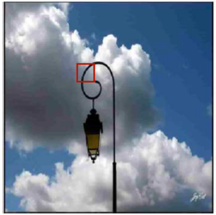
$$R_i^c(x) = \frac{\partial S_c(x)}{\partial x_i}$$



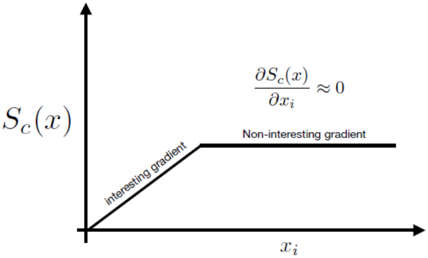
Deep Neural Networks are usually trained till "Saturation"

PERTURBING INPUTS

- Small perturbations at the saturation point do not give us interesting gradients
- Extreme perturbation (to say a baseline image) can give us interesting gradients

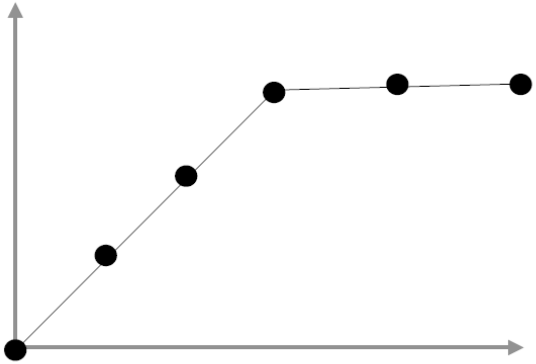


$$R_i^c(x) = \frac{\partial S_c(x)}{\partial x_i}$$

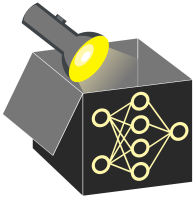


INTEGRATED GRADIENTS

Compute gradient estimate based on gradients over a path of specific perturbations

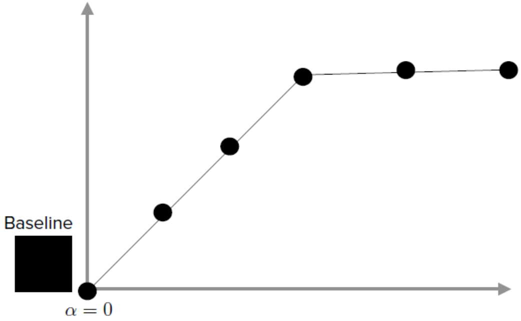


INTEGRATED GRADIENTS

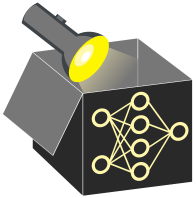


Compute gradient estimate based on gradients over a path of specific perturbations

Choose a Baseline to contrast

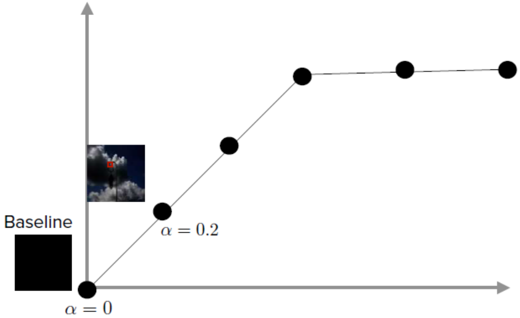


INTEGRATED GRADIENTS



Compute gradient estimate based on gradients over a path of specific perturbations

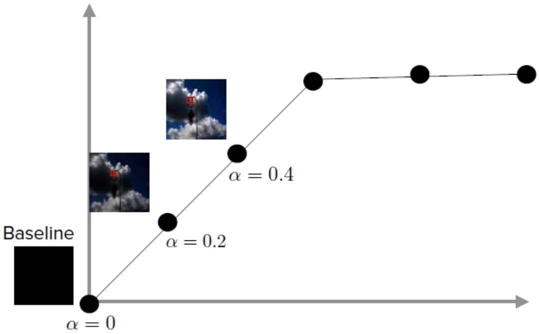
Choose a Baseline to contrast



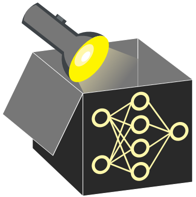
INTEGRATED GRADIENTS

Compute gradient estimate based on gradients over a path of specific perturbations

Choose a Baseline to contrast

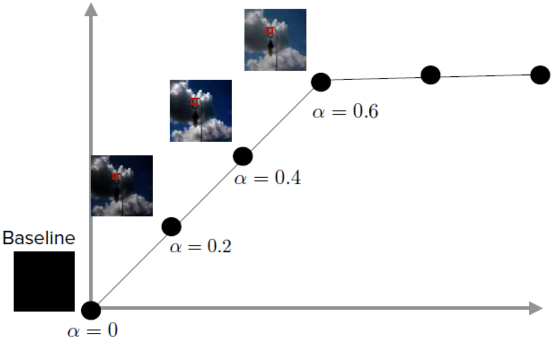


INTEGRATED GRADIENTS



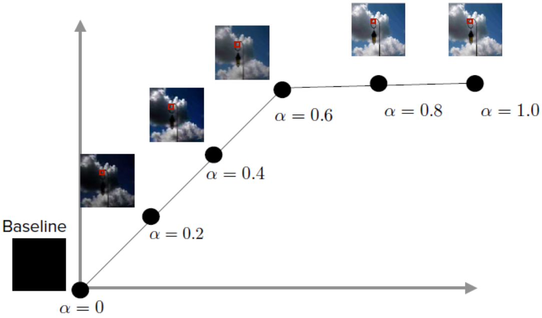
Compute gradient estimate based on gradients over a path of specific perturbations

Choose a Baseline to contrast

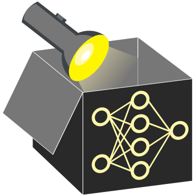


INTEGRATED GRADIENTS

Compute gradient estimate based on gradients over a path of specific perturbations
Choose a Baseline to contrast



INTEGRATED GRADIENTS



1. Choose a Baseline to contrast
2. Compute gradients at different mask values
3. Attribution = Aggregation over gradients computed for a certain set of perturbations

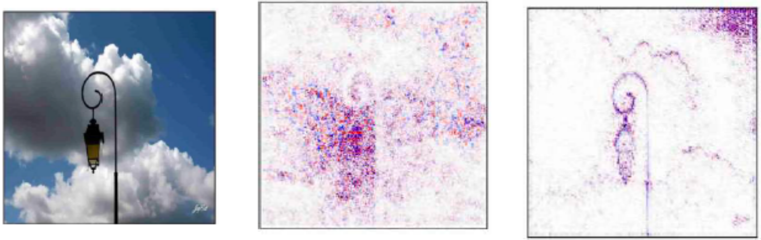
$$R_i^c(x) = x_i \cdot \int_{\alpha=0}^{1} \frac{\partial S_c(\tilde{x})}{\partial(\tilde{x}_i)} d\alpha$$

where $\tilde{x} = \overline{x} + \alpha(x - \overline{x})$

Integrated Gradients monitors how the network changes from a zero signal input to actual input through the use of gradients

BASELINE

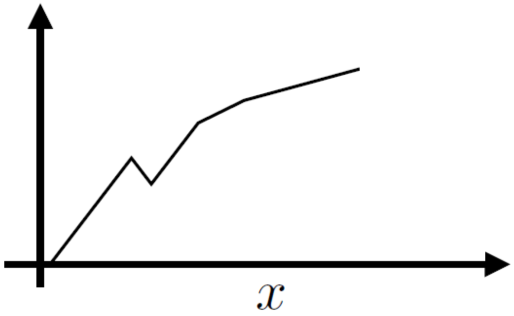
- Baseline is an information less input
- The choice of baselines matters a lot and is typically domain dependent
    - Black or gray images
    - Zero embedding in language
    - Random document in retrieval



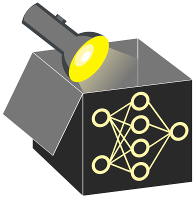
Simple Gradient   Integrated Gradients

SMOOTHGRAD

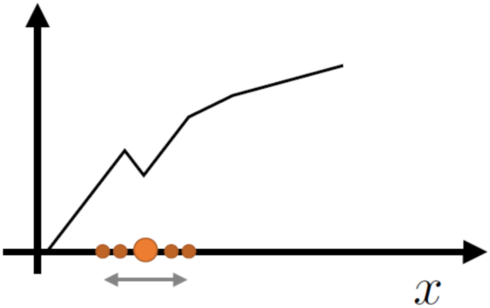
- Gradients are local ways to measure sensitivity
- In highly nonlinear loss surfaces you obtain quite noisy gradients
  - In this figure, majority of the neighbourhood gives positive gradient



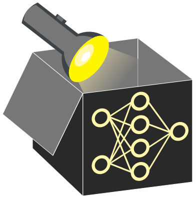
SMOOTHGRAD



- Calculate multiple copies of the input with a small noise (usually Gaussian noise)
- Actual gradient is the average of the gradients of each of the copies



CONCLUSION



- Gradients are central in computing feature attributions and are visualised using saliency maps
- Simple gradient-based approaches for neural networks attribute the importance back to the input features
- Deep learning models suffer from critical problems for gradient-based methods
  - Models are trained to saturation given near-zero gradients — Integrated Gradients
  - Gradients are unstable due to highly non-linear loss surface — SmoothGrad
- Tons of other approaches proposed in the literature
- Caution that explanations might disagree with each other
- Caution that gradient-based approaches need to be adapted depending on the input style