Bayesian Machine Learning and Uncertainty Estimation

Federico Albanese
(falbanese@dc.uba.ar)

Esteban Feuerstein

Leandro Lombardi

Instituto de Ciencias de la Computación (ICC), UBA-CONICET
Instituto de Cálculo (IC), UBA-CONICET

Esteban Feuerstein

Leandro Lombardi

Instituto de Ciencias de la Computación (ICC), UBA-CONICET
Instituto de Cálculo (IC), UBA-CONICET

Esteban Feuerstein

Leandro Lombardi

Instituto de Ciencias de la Computación (ICC), UBA-CONICET
Instituto de Cálculo (IC), UBA-CONICET

Bayesian Machine Learning

Bayesian Machine Learning

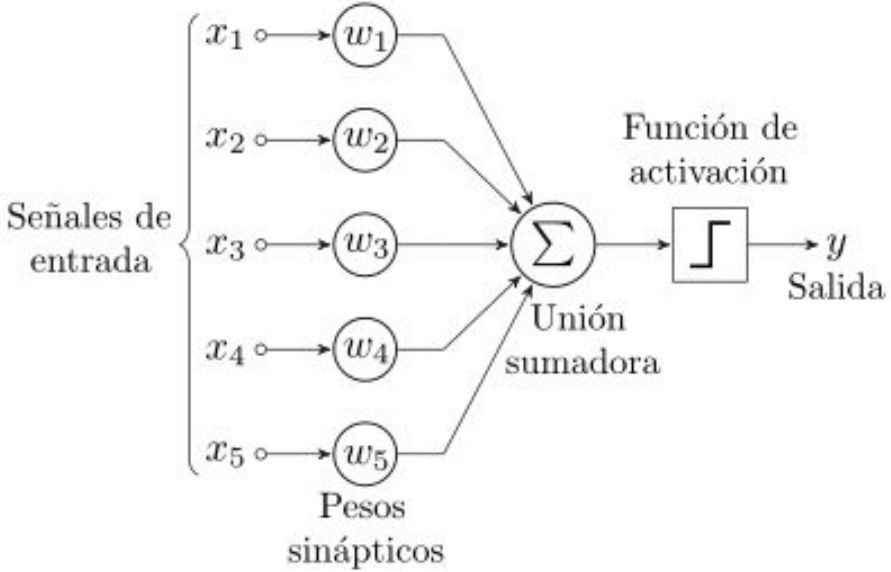
Ba[illegible]an Machine Learning

Clasificación supervisada

¿En la foto hay un gato o un perro?

Ba[illegible]an Machine Learning

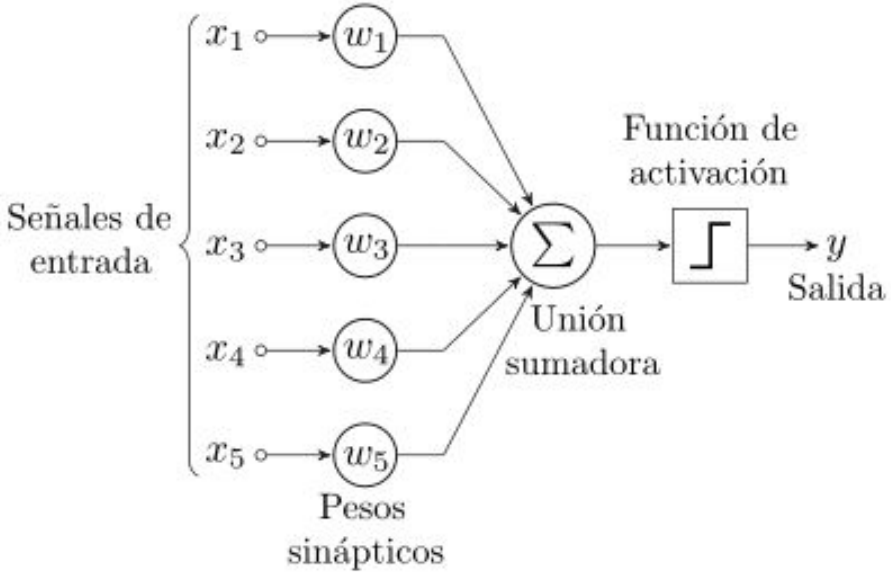
Perceptrón simple



$$Y = \sigma(\Sigma W_i * X_i + b)$$

Ba an Machine Learning

Perceptrón simple

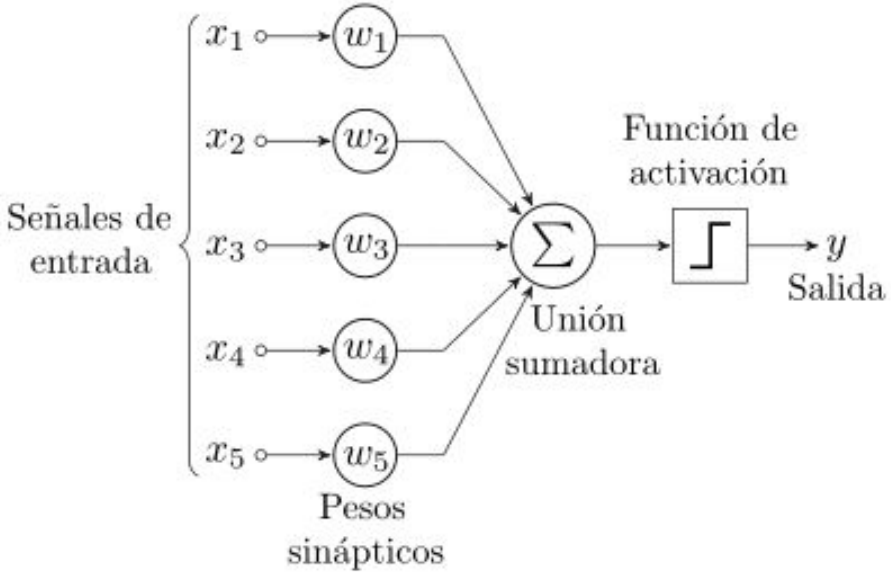


$$Y = \sigma(\Sigma W_i * X_i + b)$$

- Los ***W*** y ***b*** (sobre todo en deep learning) funcionan generalmente como una caja negra que no permite, viéndolos, saber por qué un modelo funciona o no.

Ba[illegible]an Machine Learning

Perceptrón simple

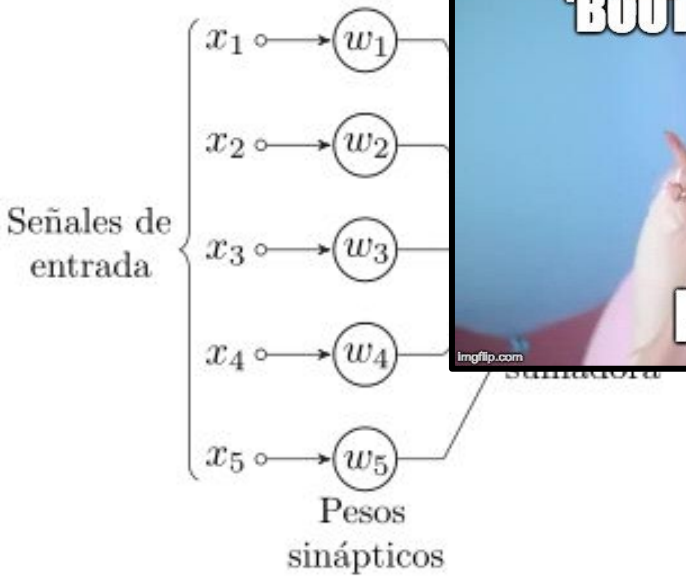


$$Y = \sigma(\Sigma W_i * X_i + b)$$

- Los ***W*** y ***b*** (sobre todo en deep learning) funcionan generalmente como una caja negra que no permite, viéndolos, saber por qué un modelo funciona o no.
- ***Y*** es un número, no un intervalo de certeza.

Ba an Machine Learning

Perceptrón simple



$$(\Sigma W_i * X_i + b)$$

- …úmero, no un intervalo de
- Los ***W*** y ***b*** (sobre todo en deep learning) funcionan generalmente como una caja negra que no permite, viéndolos, saber por qué un modelo funciona o no.

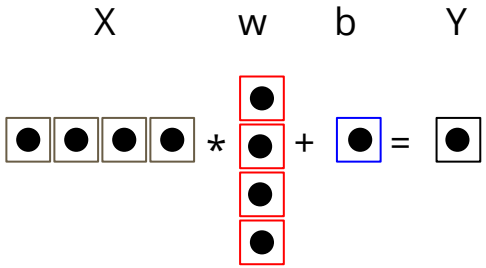
Bayesian Machine Learning

"Understanding what a **model does not know is a critical** part of many machine learning systems. Unfortunately, today's deep learning algorithms are usually unable to understand their uncertainty."

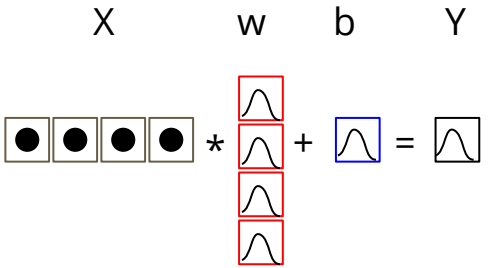
Alex Kendall; "Deep Learning Is Not Good Enough, We Need Bayesian Deep Learning for Safe AI" (2017) alexgkendall.com

Bayesian Machine Learning

No bayesiano

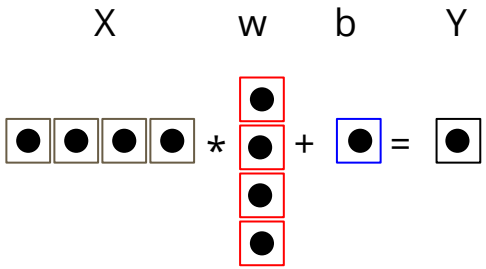


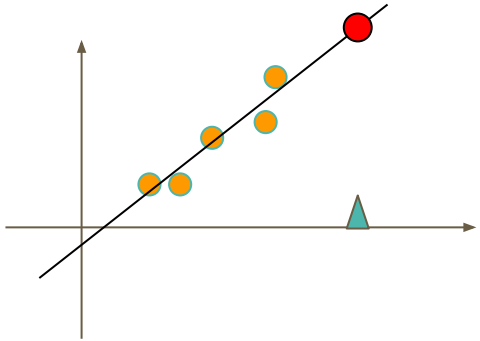
Bayesiano



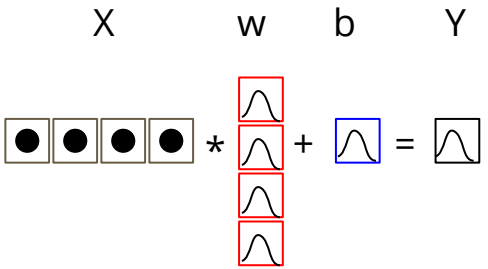
Bayesian Machine Learning

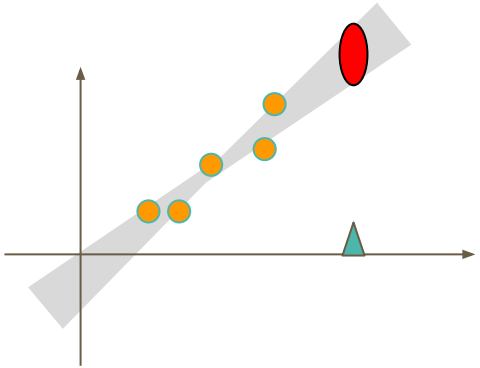
No bayesiano





Bayesiano

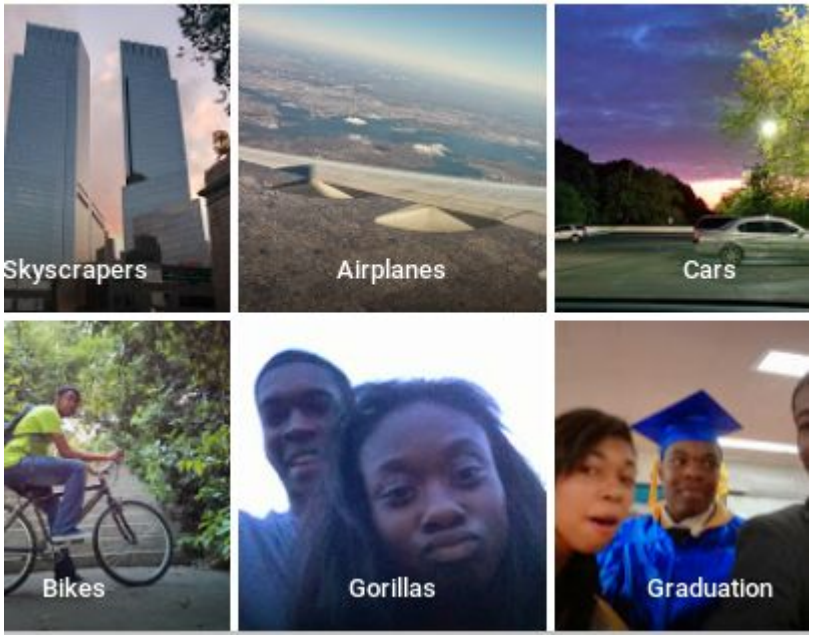




Ejemplos donde hubiese sido importante usarlo

Ejemplos donde hubiese sido importante usarlo

- Google photos etiqueto como gorilas a una foto de personas de color.



Ejemplos donde hubiese sido importante usarlo

- Google photos etiqueto como gorilas a una foto de personas de color.
- Choque de un auto Tesla (self driving).

Ejemplos donde hubiese sido importante usarlo

- Google photos etiqueto como gorilas a una foto de personas de color.
- Choque de un auto Tesla (self driving).
- Finanzas.
- Diagnostico médico.
- etc.

Tipo de incertezas

Aleatoric uncertainty

Es una incerteza relacionada con la data que el modelo no puede explicar.

Por lo tanto más data no implica a una reducción de dicha incerteza.

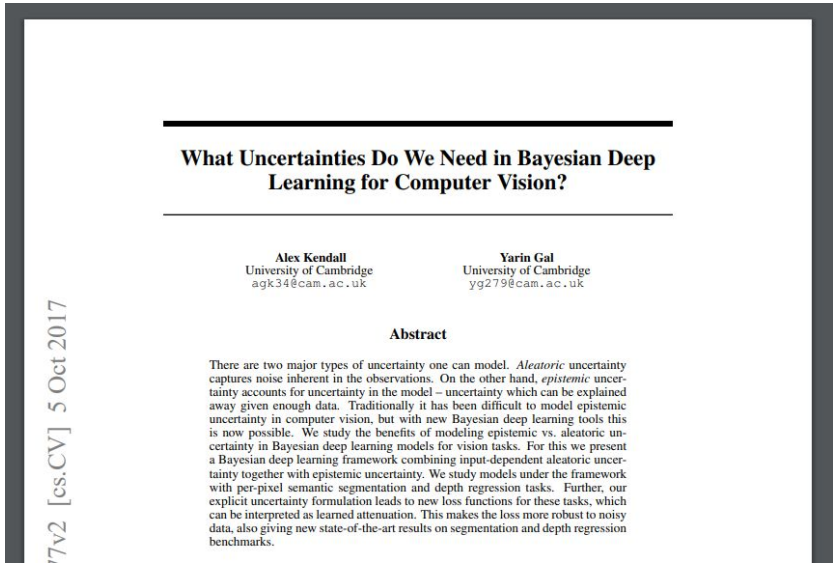
Epistemic uncertainty

Es una incerteza relacionada con el modelo que generó la predicción.

Por lo tanto más data implica que el modelo puede aprender más y reduce esta incerteza.

Tipo de incertezas

Observan que la **Aleatoric variance *no cambia*** drásticamente al modificar el dataset ni al reducirlo de tamaño. Por otro lado la **Epistemic Variance *aumenta*** considerablemente al reducir el tamaño del set de entrenamiento.



What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?

Alex Kendall
University of Cambridge
agk34@cam.ac.uk

Yarin Gal
University of Cambridge
yg279@cam.ac.uk

Abstract

There are two major types of uncertainty one can model. *Aleatoric* uncertainty captures noise inherent in the observations. On the other hand, *epistemic* uncertainty accounts for uncertainty in the model – uncertainty which can be explained away given enough data. Traditionally it has been difficult to model epistemic uncertainty in computer vision, but with new Bayesian deep learning tools this is now possible. We study the benefits of modeling epistemic vs. aleatoric uncertainty in Bayesian deep learning models for vision tasks. For this we present a Bayesian deep learning framework combining input-dependent aleatoric uncertainty together with epistemic uncertainty. We study models under the framework with per-pixel semantic segmentation and depth regression tasks. Further, our explicit uncertainty formulation leads to new loss functions for these tasks, which can be interpreted as learned attenuation. This makes the loss more robust to noisy data, also giving new state-of-the-art results on segmentation and depth regression benchmarks.

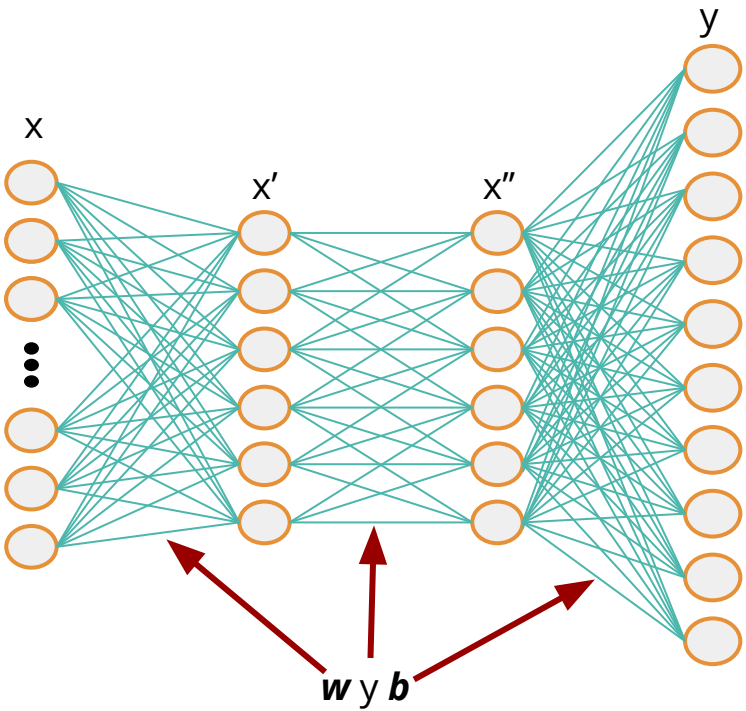
training data	Testing data	Aleatoric Variance	Epistemic Variance
dataset 1	dataset 1	0.485	2.78
25% dataset1	dataset 1	0.506	7.73
dataset 1	dataset 2	0.461	4.87
25% dataset 1	dataset 2	0.388	15.0

Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In *Advances in neural information processing systems* (pp. 5574-5584).

Dropout

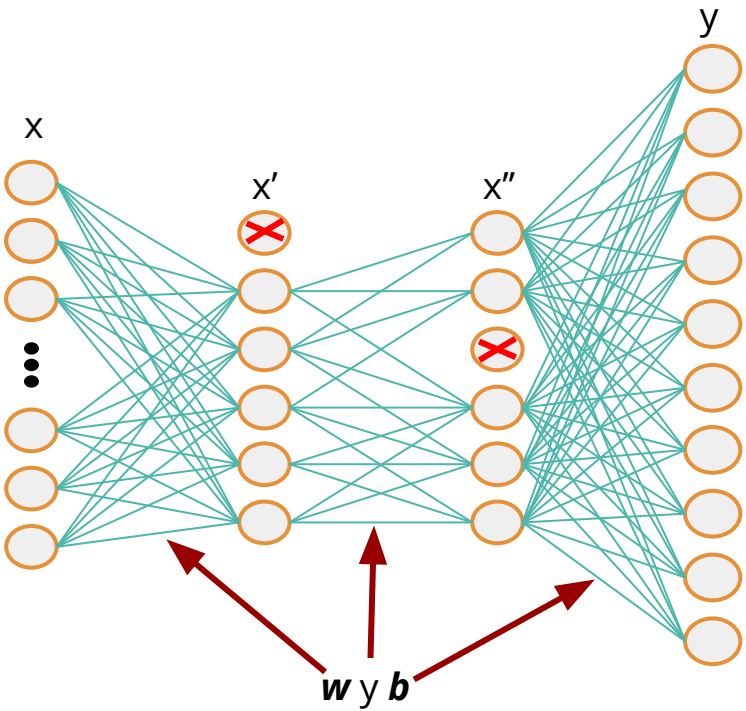
Dropout

Red Neuronal:



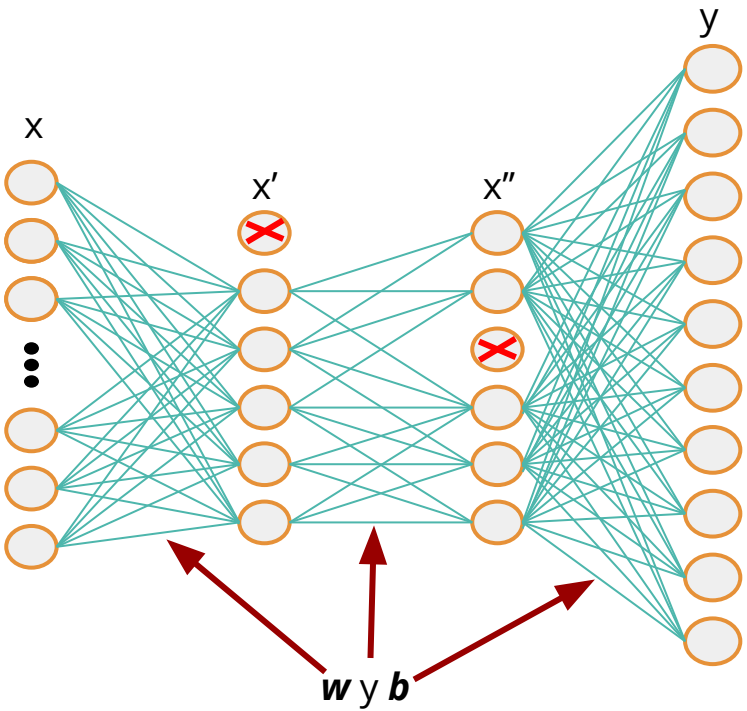
Dropout

Red Neuronal:



Dropout

Red Neuronal:

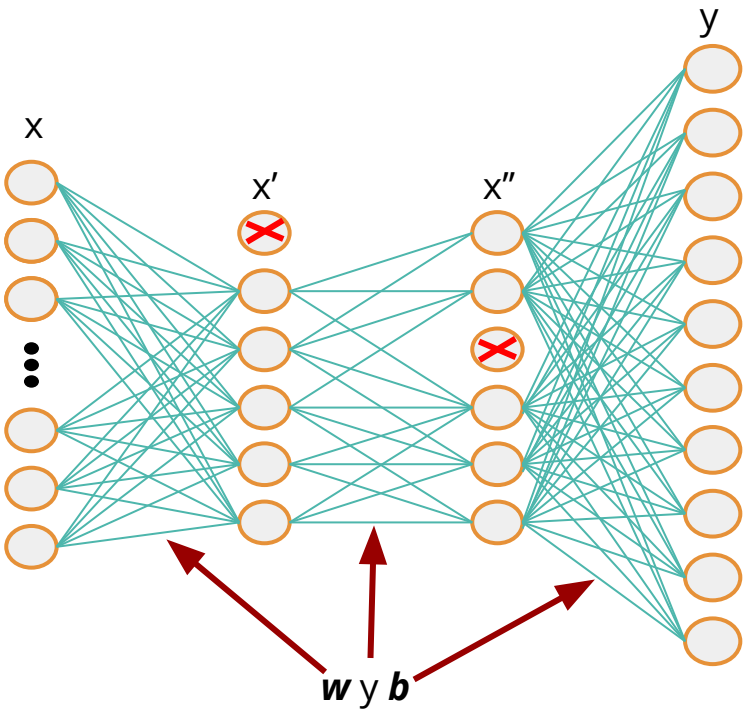


Formalmente:

- Multiplicamos por un vector binario (de 1 y 0) aleatorio para apagar algunas neuronas.
- Se define una probabilidad p de que una neurona se apague o no.

Dropout

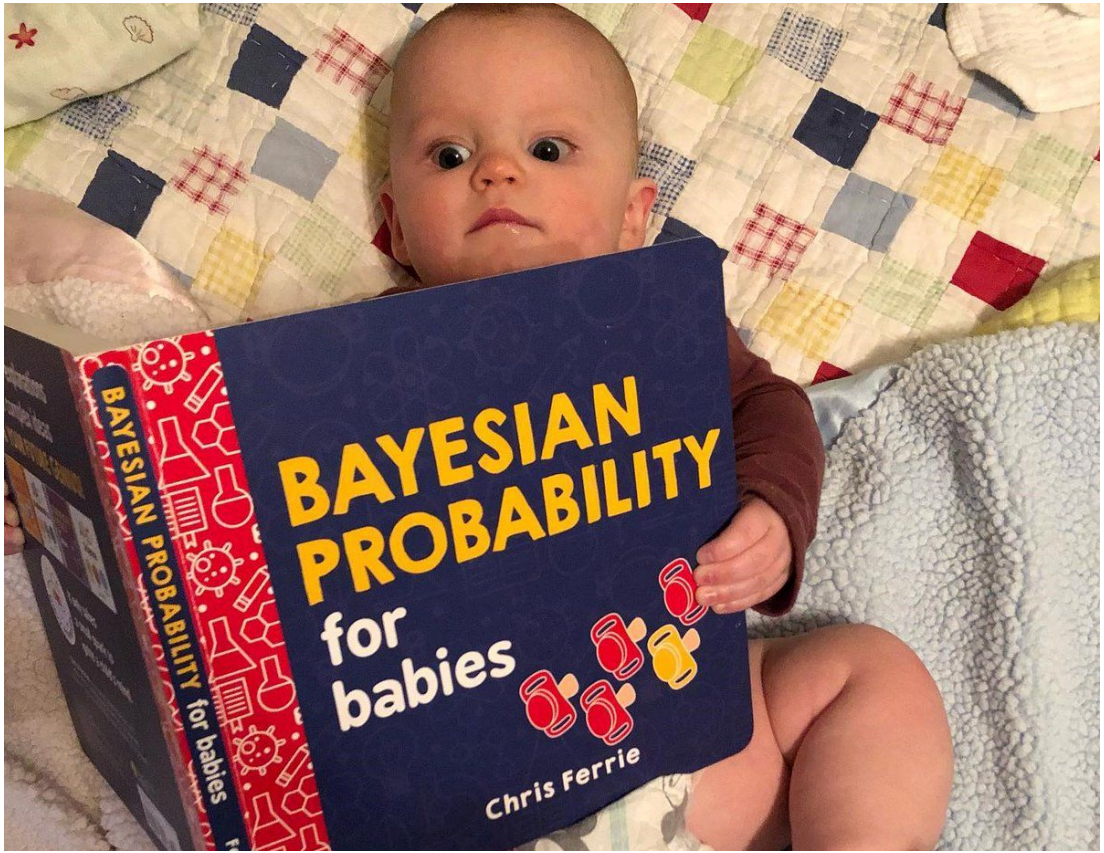
Red Neuronal:



Ventajas:

- Dropout es una regularización. Reduce el overfitting.

Bayesian Deep Learning


BAYESIAN PROBABILITY
for babies
Chris Ferrie
BAYESIAN PROBABILITY for babies

Bayesian deep Learning

- Ahora considero a los parámetros del modelo una variable aleatoria (los ***w*** y los ***b***) y cada dato que observó va alterando la distribución posterior.

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

- Ahora considero a los parámetros del modelo una variable aleatoria (los ***w*** y los ***b***) y cada dato que observó va alterando la distribución posterior.

$$p(w|X,y) = \frac{P(y|X,w)p(w)}{p(y|X)}$$

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

- Ahora considero a los parámetros del modelo una variable aleatoria (los ***w*** y los ***b***) y cada dato que observó va alterando la distribución posterior.

$$p(w|X,y) = \frac{P(y|X,w)p(w)}{p(y|X)}$$

- Lo que yo quiero saber es y* para un x* (una nueva predicción):

$$p(y^*|x^*,X,y) = \int p((y^*|x^*,w)p(w|X,y)dw$$

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

- Ahora considero a los parámetros del modelo una variable aleatoria (los ***w*** y los ***b***) y cada dato que observó va alterando la distribución posterior.

$$p(w|X,y) = \frac{P(y|X,w)p(w)}{p(y|X)}$$

- Lo que yo quiero saber es y* para un x* (una nueva predicción):

$$p(y^*|x^*,X,y) = \int p((y^*|x^*,w)p(w|X,y)dw$$

Se busca aproximar p(w|X,y).

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

Busco aproximar $p(w|X,y)$ por una función $q_\theta(w)$.

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

Busco aproximar p(w|X,y) por una función $q_\theta(w)$.

En otras palabras, busco minimizar la divergencia de Kullback–Leibler:

$$L(\theta) = KL(q_\theta(W), p(W|X, y))$$

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

Busco aproximar p(w|X,y) por una función $q_\theta(w)$.

En otras palabras, busco minimizar la divergencia de Kullback–Leibler:

$$L(\theta) = KL(q_\theta(W), p(W|X, y))$$

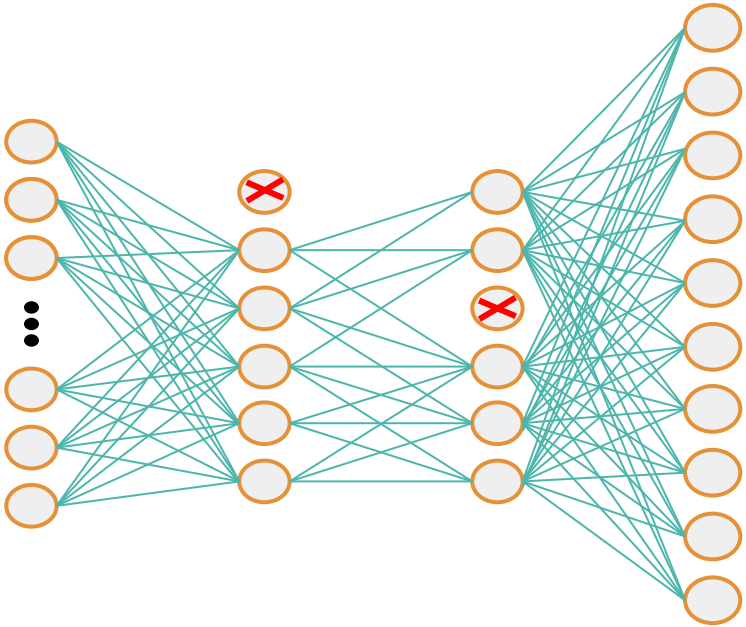
En el paper de Gal y Ghahramani (2016) demuestran que la función objetivo de inferencia variacional (la ecuación de arriba) es equivalente a la función objetivo de MC dropout reescribiendo algunos términos.

Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Bayesian deep Learning

¿Cómo lo aplico?

- Uso dropout durante la evaluación del modelo generando múltiples outputs. (Evaluo múltiples veces el mismo input haciendo 0 distintas neuronas. Muestreo W usando una estimación de Monte Carlo a partir de los W que tengo.)



Obtener un conjunto de predicciones:

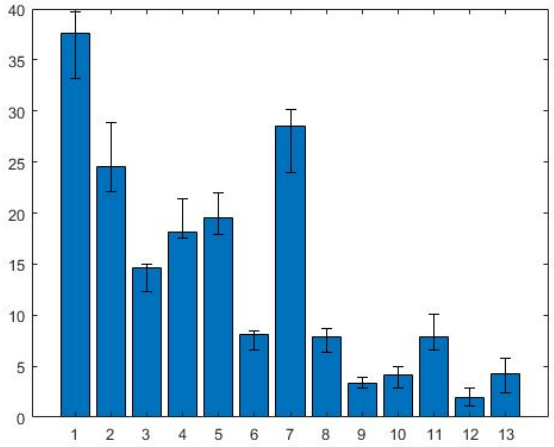
$[y_1, y_2, ...]$

Bayesian deep Learning

¿Cómo lo aplico?

- Uso dropout durante la evaluación del modelo generando múltiples outputs. (Evaluo múltiples veces el mismo input haciendo 0 distintas neuronas. Muestreo W usando una estimación de Monte Carlo a partir de los W que tengo.)

- Calculo la media y la STD de los outputs para saber la incerteza de la predicción.



Incerteza
de las incertezas

Incerteza de las incertezas

¿Cómo saber si MC dropout está calculando correctamente las incertezas y es un método válido para nuestra tarea en particular?

¿Cómo sé cuál es una incerteza correcta?

Incerteza de las incertezas

Pasos:

- Entrenar un modelo que resuelva la tarea.

Incerteza de las incertezas

Pasos:

- Entrenar un modelo que resuelva la tarea.
- Simulación:
  - Generar un dataset perturbando la primera capa de la red y quedarse con varios outputs (Para cada x_i se obtienen varios y_i^t)

Incerteza de las incertezas

Pasos:

- Entrenar un modelo que resuelva la tarea.
- Simulación:
  - Generar un dataset usando first layer dropout y quedarse con varios outputs
    (Para cada x_i se obtienen varios y_i^t)
  - A partir de los y_i^t puedo calcular una desviación estándar y una media.
    (Si los y_i^t son probabilidades, se puede usar una distribución de Bernoulli para calcular los labels)

Incerteza de las incertezas

Pasos:

- Entrenar un modelo que resuelva la tarea.
- Simulación:
  - Generar un dataset usando first layer dropout y quedarse con varios outputs
    (Para cada x_i se obtienen varios y_i^t)
  - A partir de los y_i^t puedo calcular una desviación estándar y una media.
    (Si los y_i^t son probabilidades, se puede usar una distribución de Bernoulli para calcular los labels)
  - Entrenar nuevos modelos con MC dropout sobre el nuevo dataset.

Incerteza de las incertezas

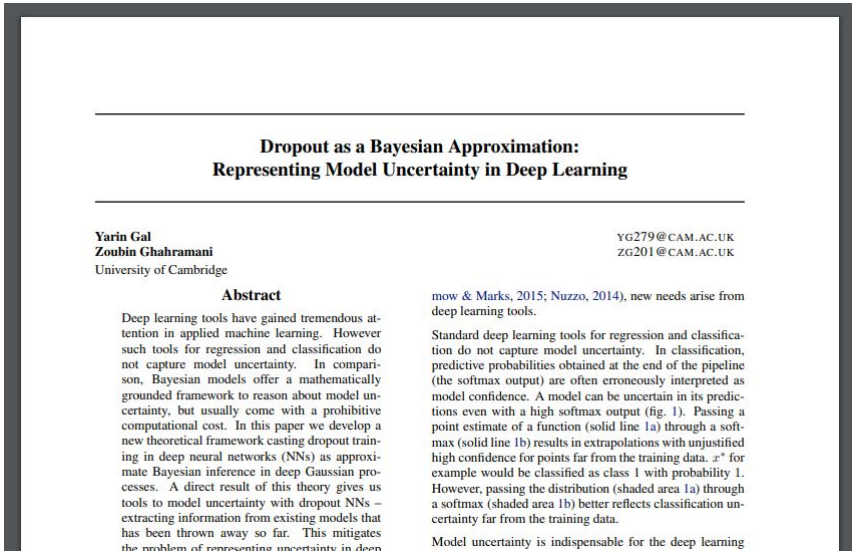
Pasos:

- Entrenar un modelo que resuelva la tarea.
- Simulación:
  - Generar un dataset usando first layer dropout y quedarse con varios outputs
    (Para cada x_i se obtienen varios y_i^t)
  - A partir de los y_i^t puedo calcular una desviación estándar y una media.
    (Si los y_i^t son probabilidades, se puede usar una distribución de Bernoulli para calcular los labels)
  - Entrenar nuevos modelos con MC dropout sobre el nuevo dataset.
- Comparar la desviación estándar de los nuevos modelos con la del modelo original.

Aplicaciones

Aplicaciones: Reinforcement Learning

En una tarea de exploración, un agente logra explorar **más rápido** espacio 2D en donde hay rewards positivos y negativos usando dropout Q-network en vez de una estrategia epsilon greedy.



Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning

Yarin Gal YG279@CAM.AC.UK
Zoubin Ghahramani ZG201@CAM.AC.UK
University of Cambridge

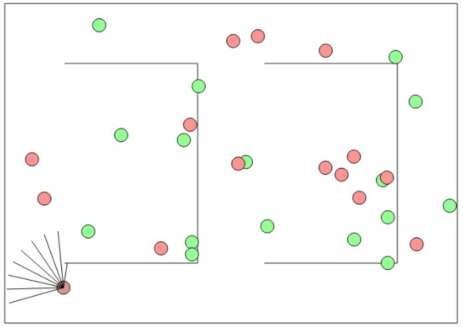
Abstract

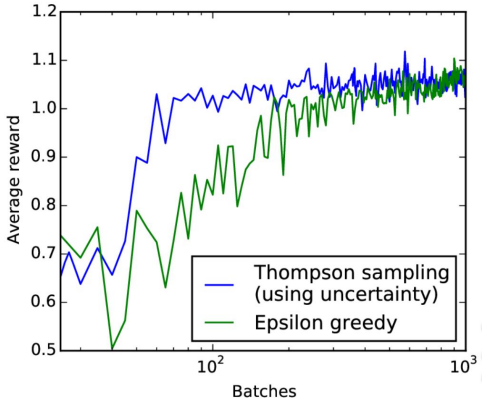
Deep learning tools have gained tremendous attention in applied machine learning. However such tools for regression and classification do not capture model uncertainty. In comparison, Bayesian models offer a mathematically grounded framework to reason about model uncertainty, but usually come with a prohibitive computational cost. In this paper we develop a new theoretical framework casting dropout training in deep neural networks (NNs) as approximate Bayesian inference in deep Gaussian processes. A direct result of this theory gives us tools to model uncertainty with dropout NNs – extracting information from existing models that has been thrown away so far. This mitigates the problem of representing uncertainty in deep

mow & Marks, 2015; Nuzzo, 2014), new needs arise from deep learning tools.

Standard deep learning tools for regression and classification do not capture model uncertainty. In classification, predictive probabilities obtained at the end of the pipeline (the softmax output) are often erroneously interpreted as model confidence. A model can be uncertain in its predictions even with a high softmax output (fig. 1). Passing a point estimate of a function (solid line 1a) through a softmax (solid line 1b) results in extrapolations with unjustified high confidence for points far from the training data. x^* for example would be classified as class 1 with probability 1. However, passing the distribution (shaded area 1a) through a softmax (shaded area 1b) better reflects classification uncertainty far from the training data.

Model uncertainty is indispensable for the deep learning

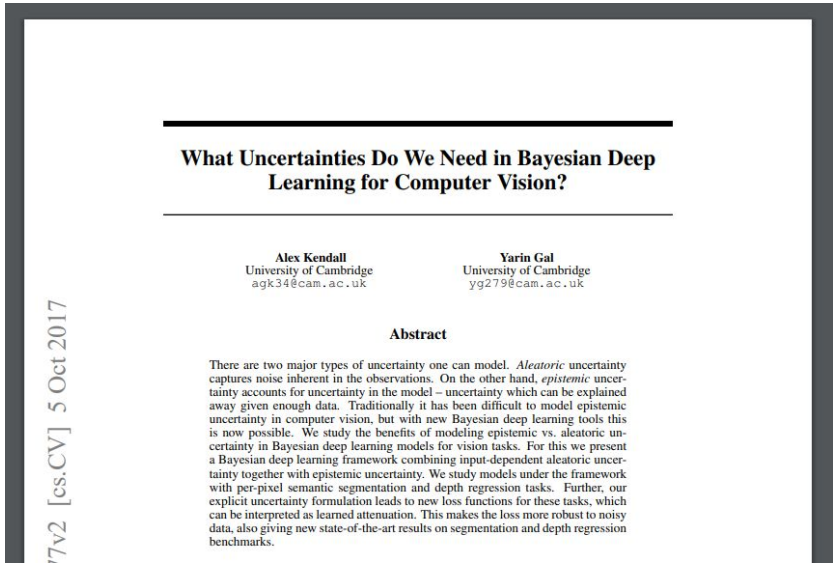




Gal, Y., & Ghahramani, Z. (2015). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *arXiv preprint arXiv:1506.02142*.

Aplicaciones: Multitask learning

Usan una loss function que tiene en cuenta la incerteza de las predicciones para distintas tareas al momento de evaluar la loss.



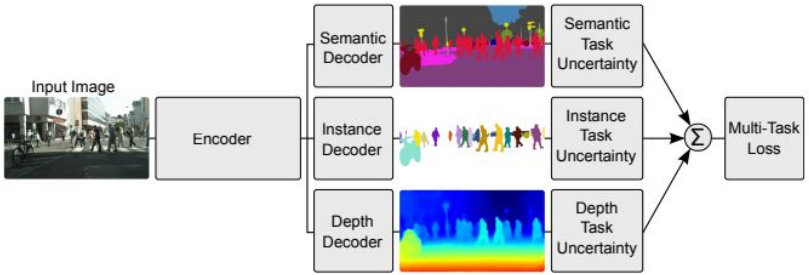
What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?

Alex Kendall
University of Cambridge
agk34@cam.ac.uk

Yarin Gal
University of Cambridge
yg279@cam.ac.uk

Abstract

There are two major types of uncertainty one can model. *Aleatoric* uncertainty captures noise inherent in the observations. On the other hand, *epistemic* uncertainty accounts for uncertainty in the model – uncertainty which can be explained away given enough data. Traditionally it has been difficult to model epistemic uncertainty in computer vision, but with new Bayesian deep learning tools this is now possible. We study the benefits of modeling epistemic vs. aleatoric uncertainty in Bayesian deep learning models for vision tasks. For this we present a Bayesian deep learning framework combining input-dependent aleatoric uncertainty together with epistemic uncertainty. We study models under the framework with per-pixel semantic segmentation and depth regression tasks. Further, our explicit uncertainty formulation leads to new loss functions for these tasks, which can be interpreted as learned attenuation. This makes the loss more robust to noisy data, also giving new state-of-the-art results on segmentation and depth regression benchmarks.



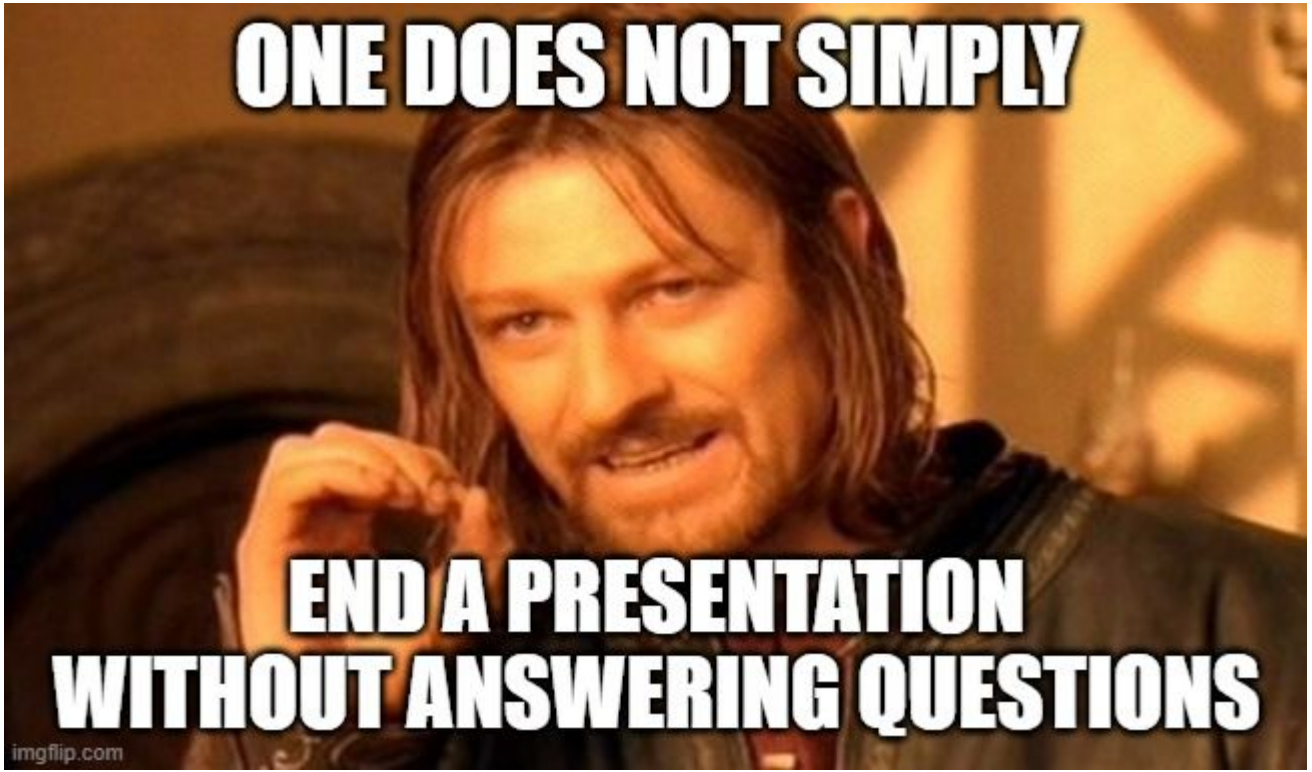
$$L = \frac{1}{2\sigma_1^2} L_1(W) + \frac{1}{2\sigma_2^2} L_2(W) + log(\sigma_1 \sigma_2)$$

Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In *Advances in neural information processing systems* (pp. 5574-5584).

Conclusiones

- Bayesian deep Learning permite estimar la incerteza de las predicciones.
- Saber la incerteza de las predicciones de nuestros modelos es vital para su uso en producción.
- Monte Carlo Dropout es una técnica simple y útil para estimar incertezas.

Preguntas


ONE DOES NOT SIMPLY
END A PRESENTATION
WITHOUT ANSWERING QUESTIONS
imgflip.com