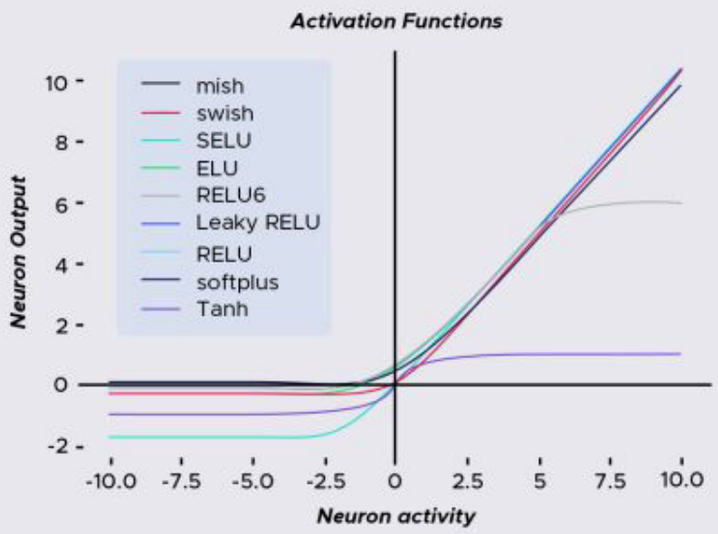
Cheat Sheet **Keras :** Fonctions et métriques

Fonctions d'activations

tensorflow.keras.activations

relu
sigmoid
softmax
softplus
softsign
tanh
SELU
elu
exponential



Exemples

```
import numpy as np
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
data = np.random.random((1000,100))
labels = np.random.randint(2,size=(1000,1))
model = Sequential()
model.add(Dense(32, activation='relu', input_dim=100))
model.add(Dense(1, activation='sigmoid'))
model.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['accuracy'])
model.fit(data, labels, epochs = 10, batch_size=32)
predictions = model.predict(data)
```

Fonctions de perte

Pertes probabilistes

binary_crossentropy
categorical_crossentropy
sparse_categorical_crossentropy
poisson
kl_divergence

Pertes de régression

mean_squared_error
mean_squared_absolute_error
mean_squared_percentage_error
mean_squared_logarithmic_error
cosine_similarity
huber
log_cosh

Hinge Losses - Marge maximale

hinge
squared_hinge
categorical_hinge

Métriques : évaluer les performances du modèle

Précision (fréquence des bonnes predictions)

Accuracy
BinaryAccuracy
CategoricalAccuracy
TopKCategoricalAccuracy
SparseTopKCategoricalAccuracy

Métriques de régression

MeanSquaredError
RootMeanSquaedError
MeanAbsoluteError
MeanAbsolutePercentageError
MeanSquaredlogarithmicError
CosineSimilarity
LogCoshError

Hinge Losses - Marge maximale

AUC
Precision
Recall
True Positives
TrueNegatives
FalsePositives
FalseNegatives
PrecisionAtRecall
SensitivityAtSpecificity
SpecificityAtSensitivity

Métriques probabilistes (calcul de la crossentropie)

BinaryCrossentropy
CategoricalCrossentropy
SparseCategoricalCrossentropy
KLDivregence
Poisson

Métriques : hinge - 'maximum-margin'

Hinge
SquaredHinge
CategoricalHinge

Métrique de segmentation d'image

(IOU = true_positive J (true_positive + false_positive + false_negative)

MeanIoU