Algorithm selection and hyper-parameter optimization based quantification of error contribution in image classification pipelines

Introduction

- Quantification of contribution in terms of classification error from different components of image classification pipeline - steps, algorithms, hyper-parameters.

- Provides data scientists and domain experts with insights about the pipeline in terms of which components are important for the performance of the pipeline.

- Hyper-parameter optimization methods and algorithms to quantify error contributions - grid search, random search, Bayesian optimization.

- Random search of configurations is able to efficiently compute the error contributions.

Data analytic pipeline

Representation of a machine learning pipeline. This is represented as a generalized directed acyclic graph. S_i represents the i-th computational step in the pipeline and A_{ij} represents the j-th algorithm in the i-th step. X is the input dataset and Y is the evaluation metric.

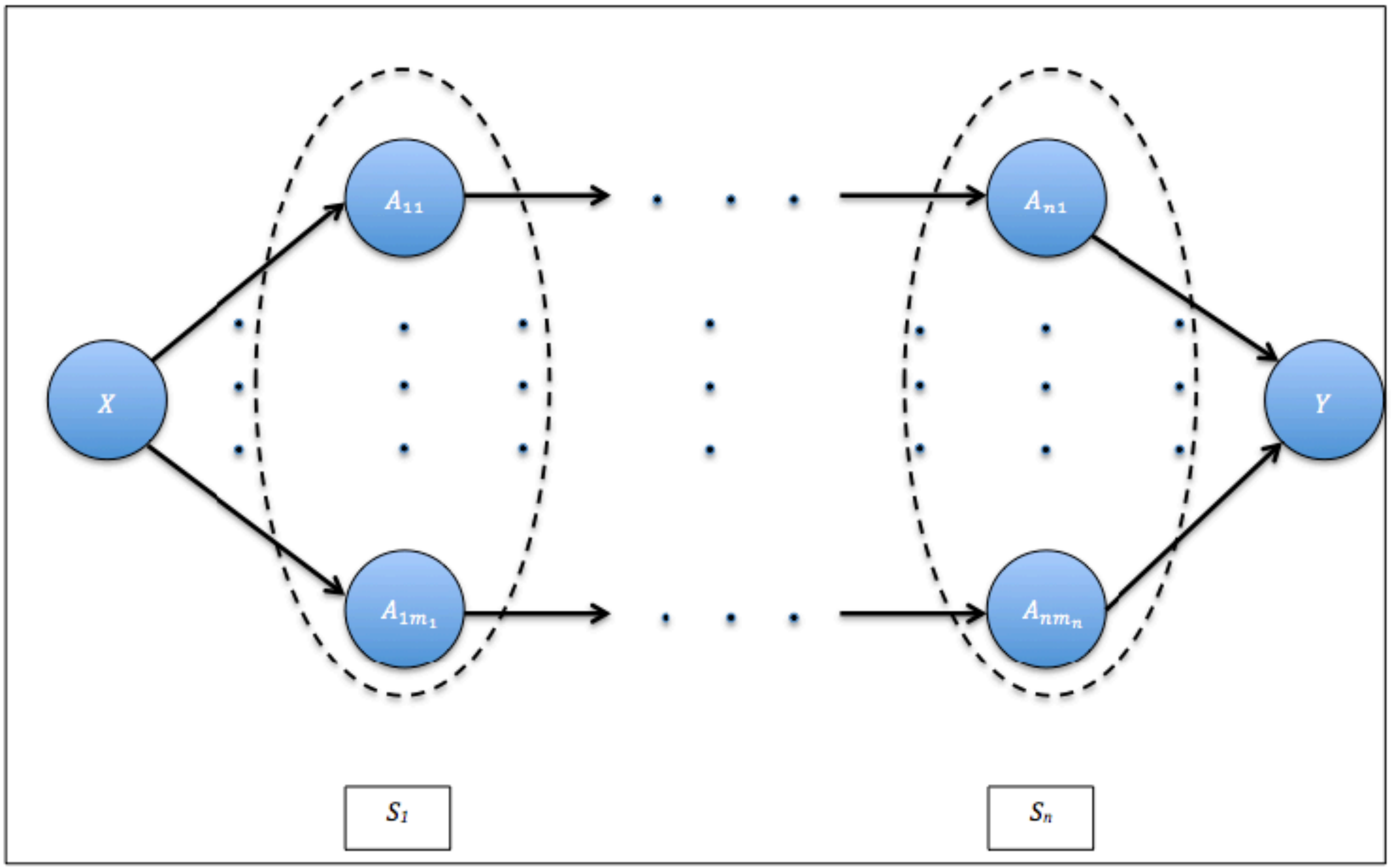
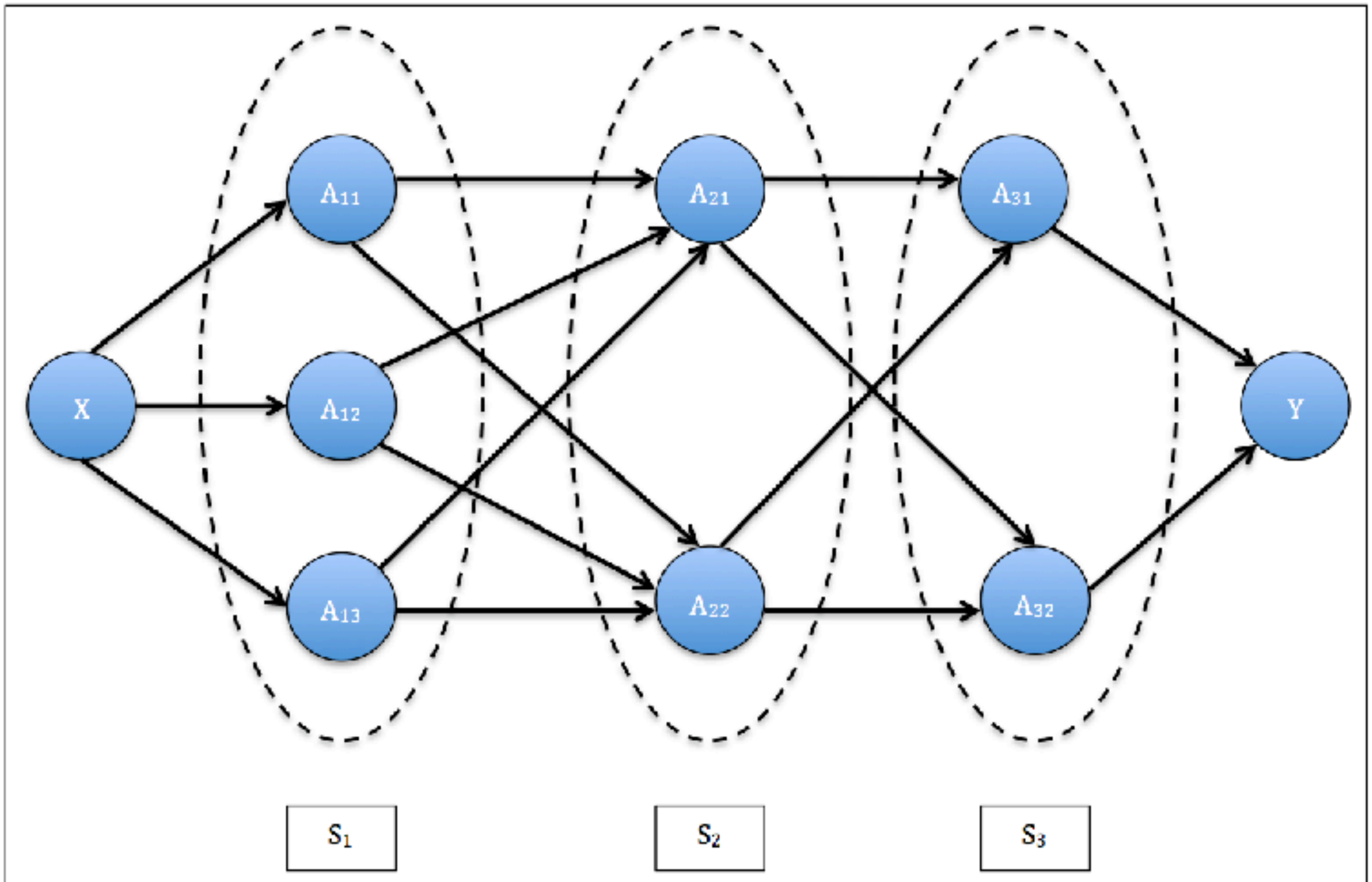


Image classification pipeline used in problem

Representation of the image classification pipeline as a directed acyclic graph used in this work. This is an instantiation of the generalized data analytic pipeline

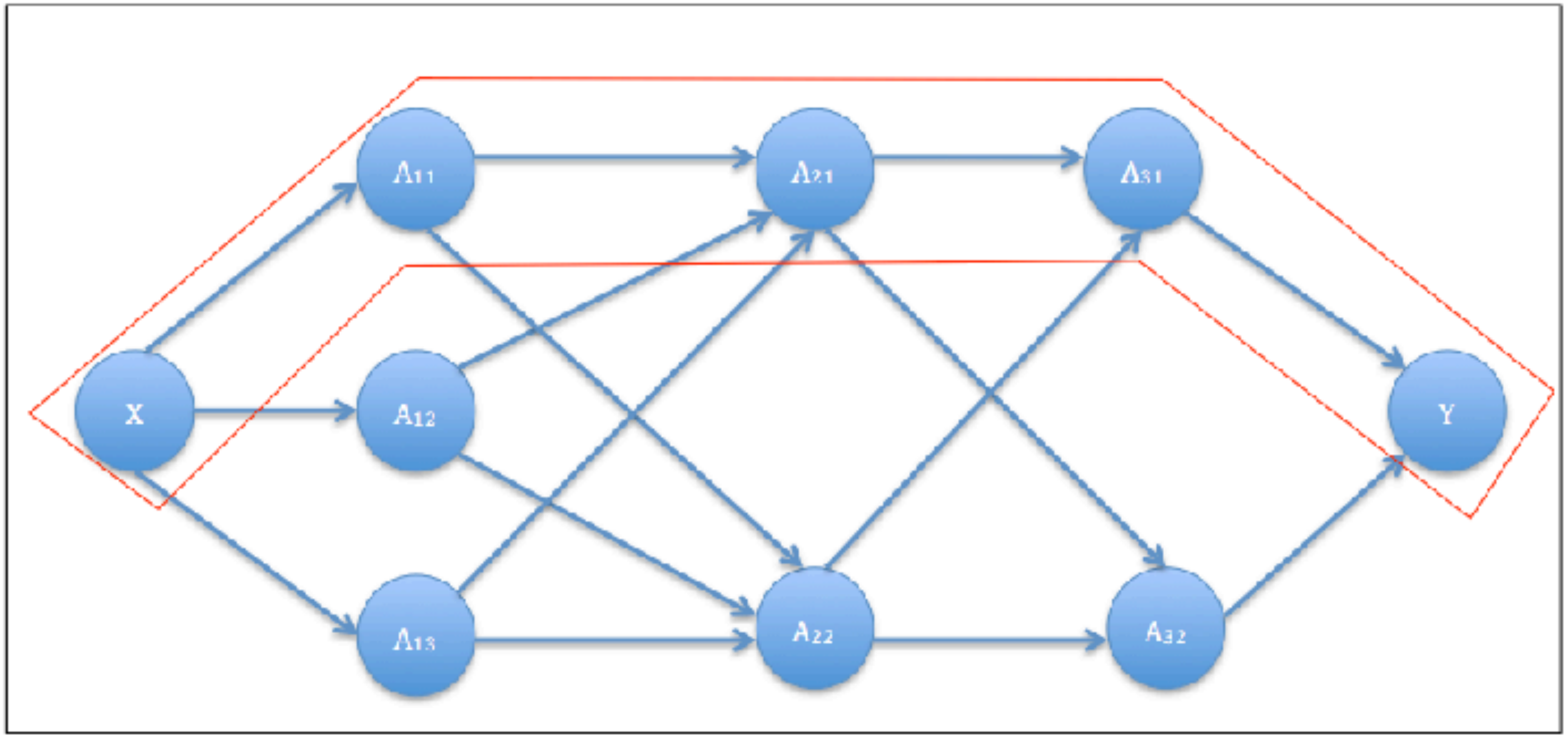


Hyper-parameter optimization (HPO)

Let the n hyperparameters in a path be denoted as $\theta_1, \theta_2, ..., \theta_n$, and let $\Theta_1, \Theta_2, ..., \Theta_n$ be their respective domains. The hyperparameter space of the path is $\Theta = \Theta_1 \times \Theta_2 \times ... \times \Theta_n$.

When trained with $\theta \in \Theta$ on data D_{train}, the validation error is denoted as $\mathcal{L}(\theta, D_{train}, D_{valid})$. Using k-fold cross-validation, the hyperparameter optimization problem for a dataset D is to minimize:

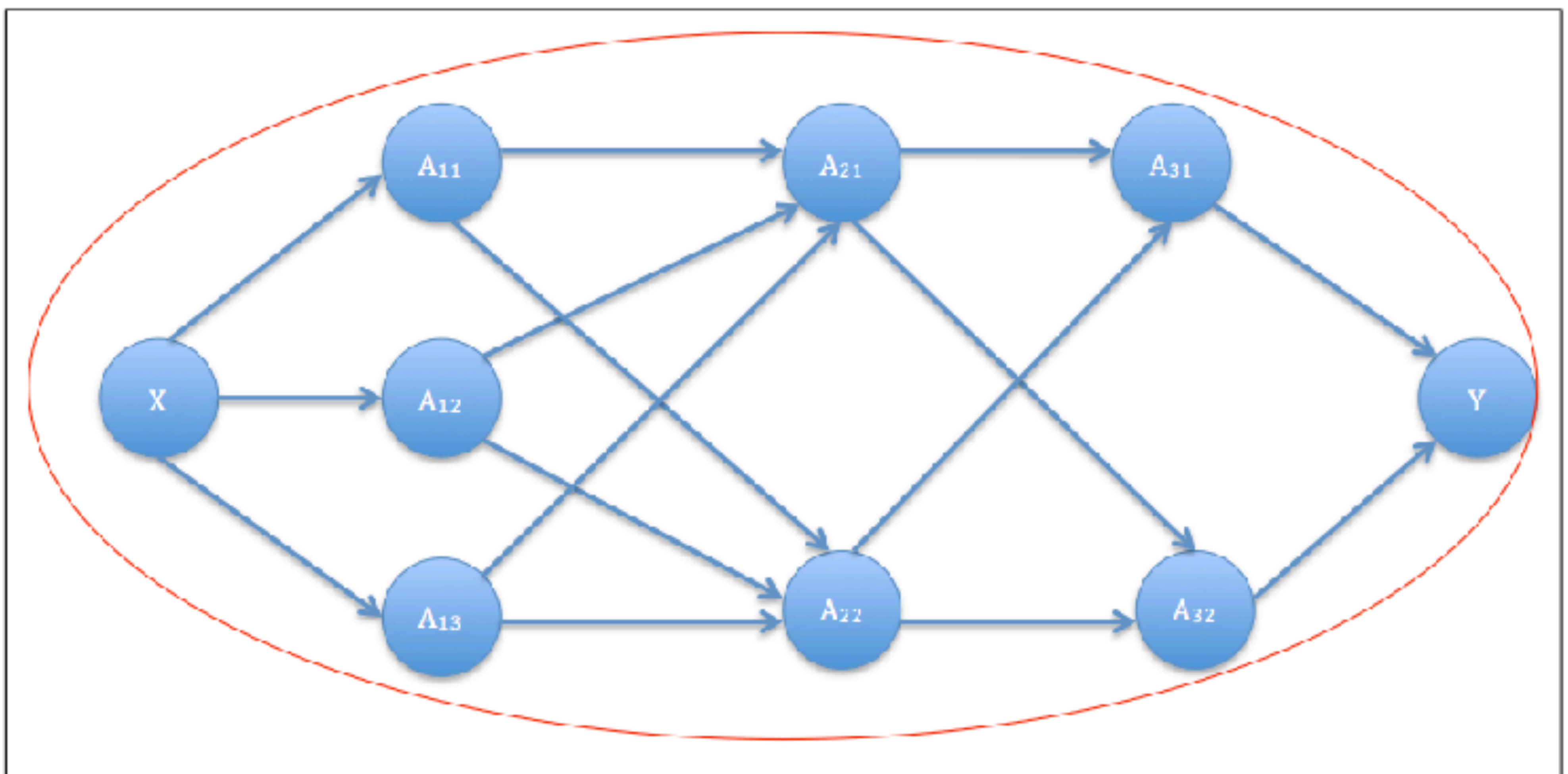
$$f^D(\theta) = \frac{1}{k} \sum_{i=1}^{k} \mathcal{L}(\theta, D_{train}^{(i)}, D_{valid}^{(i)})$$



Combined algorithm selection and hyperparameter optimization (CASH)

Let there be n computational steps in the pipeline. Each step i in the pipeline consists of algorithms $A_i(\Theta_i)$, where $A_i(\Theta_i) = \{A_{i1}(\theta_{i1}), ..., A_{im_i}(\theta_{im_i})\}$, m_i is the number of algorithms in step i, A_{ij} represents the j-th algorithm in step i, and θ_{ij} represents the set of hyperparameters corresponding to A_{ij}. The entire space of algorithms and hyperparameters is therefore given by $\mathcal{A} = A_1(\Theta_1) \times A_2(\Theta_2) \times ... \times A_n(\Theta_n)$. The objective function to be minimized for CASH is given by

$$f^D(A) = \frac{1}{k} \sum_{i=1}^{k} \mathcal{L}(A, D^{(i)}_{train}, D^{(i)}_{valid})$$



Experimental settings

Pipeline

Step	$A_{ij}(\theta_{ij})$	Definition
Feature extraction	$A_{11}(\theta_{11})$	Haralick texture features (*distance*)
	$A_{12}(\theta_{12})$	Pre-trained CNN trained on ImageNet database with VGG16 network
	$A_{13}(\theta_{13})$	Pre-trained CNN trained on ImageNet database with Inception network
Feature transformation	$A_{21}(\theta_{21})$	PCA (*whitening*)
	$A_{22}(\theta_{22})$	ISOMAP (*number of neighbors, number of components*)
Learning algorithms	$A_{31}(\theta_{31})$	Random forests (*number of trees, maximum features*)
	$A_{32}(\theta_{32})$	SVM (C, γ)

Datasets

Dataset (notation)	Distribution of classes
Breast cancer (*breast*)	*benign*: 151, *in-situ*: 93, *invasive*: 202
Brain cancer (*brain*)	*glioma*: 16, *healthy*: 210, *inflammation*: 107
Material science 1 (*matsc1*)	*dendrites*: 441, *non-dendrites*: 132
Material science 2 (*matsc2*)	*transverse*: 393, *longitudinal*: 48

Methods and algorithms for HPO and CASH

- Grid- search

- Random search

- Bayesian optimization (Sequential model agnostic configurations)

Error contribution

Agnostic methodology for quantification of error contributions

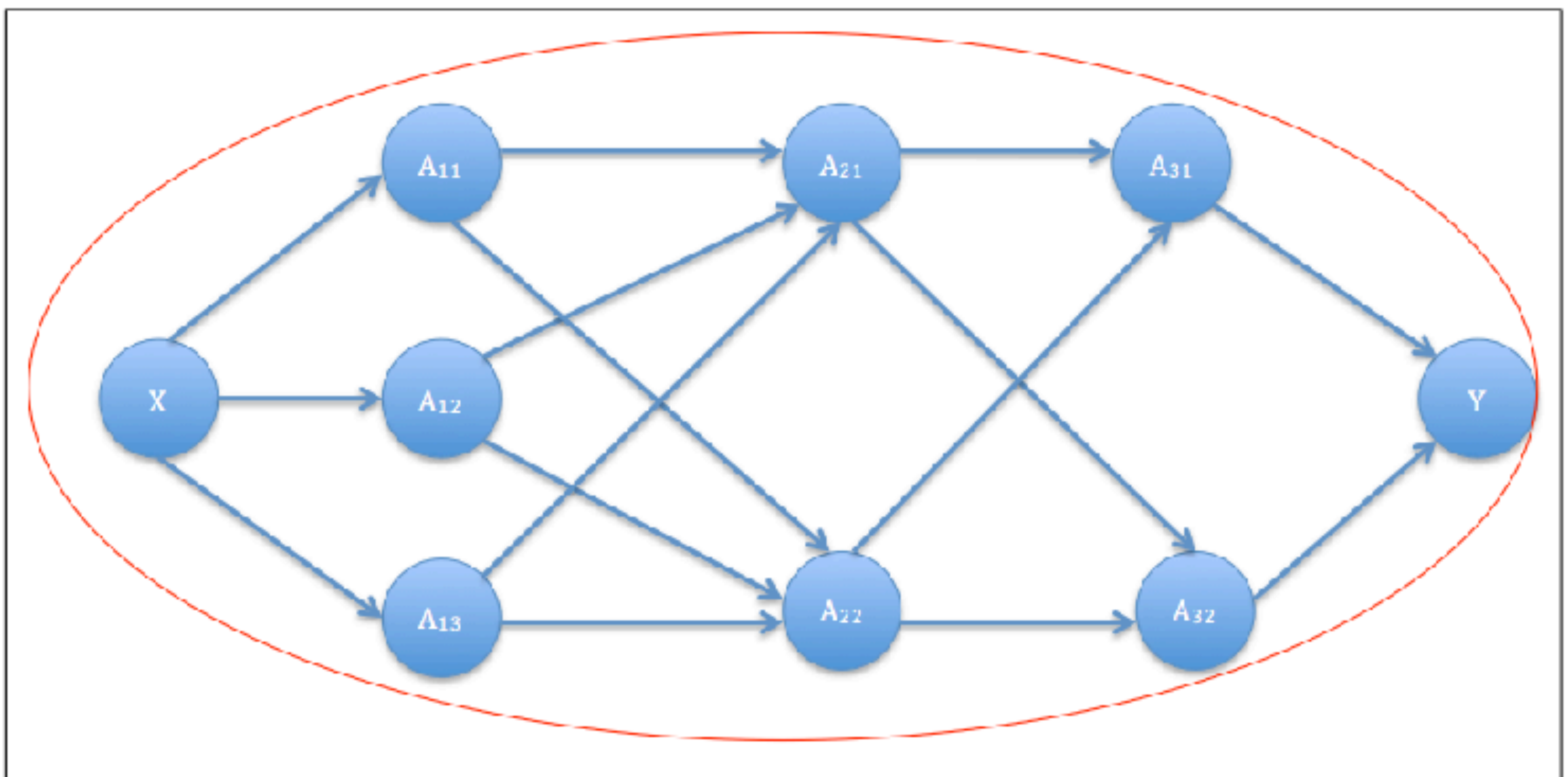
- *Agnostic* means to randomly select configurations from a particular component (algorithms, hyper-parameters) of the pipeline (CASH) or path (HPO) and optimize everything else.

- The difference between the *agnostic* error and the optimum (minimum) error provides an estimate of the error contribution from each component.

Error contribution from computational steps

Let n be the number of steps in the pipeline. Each step in the pipeline is denoted as S_i. $|S_i|$ is the number of algorithms in step i. A_{ij} denotes the j-th algorithm in the i-th step. E^* represents the minimum validation error found after optimization of the entire pipeline (using the CASH framework). $E^*_{A_{ij}}$ is the minimum validation error found with A_{ij} as the only algorithm in step i. For $i = 1, ..., n, j = 1, ..., |S_i|$,

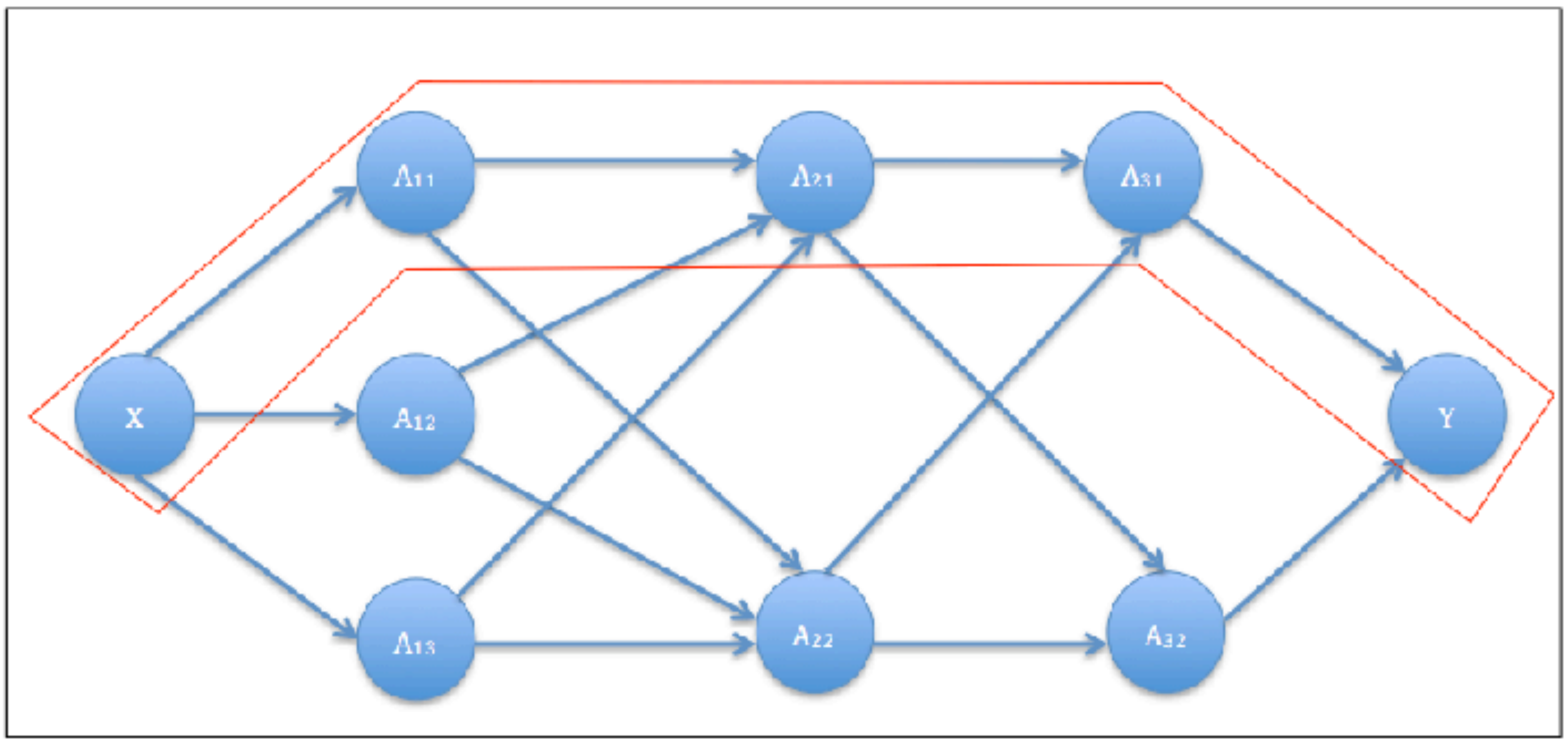
$$EC^*_{S_i} = \frac{1}{|S_i|} \sum_{z=1}^{|S_i|} E^*_{A_{ij}} - E^*,$$



Error contribution from algorithms

For, $i = 1, ..., n, j = 1, ..., |\theta_{ij}|$, $|\theta_{ij}|$ represents the number of hyperparametric configurations of A_{ij}, ${E^z_{A_{ij}}}^*$ is the minimum error obtained with the z-th configuration of θ_{ij} and $E^*_{A^p_{ij}}$ is the minimum error found over the path p that consists of algorithm A_{ij}.

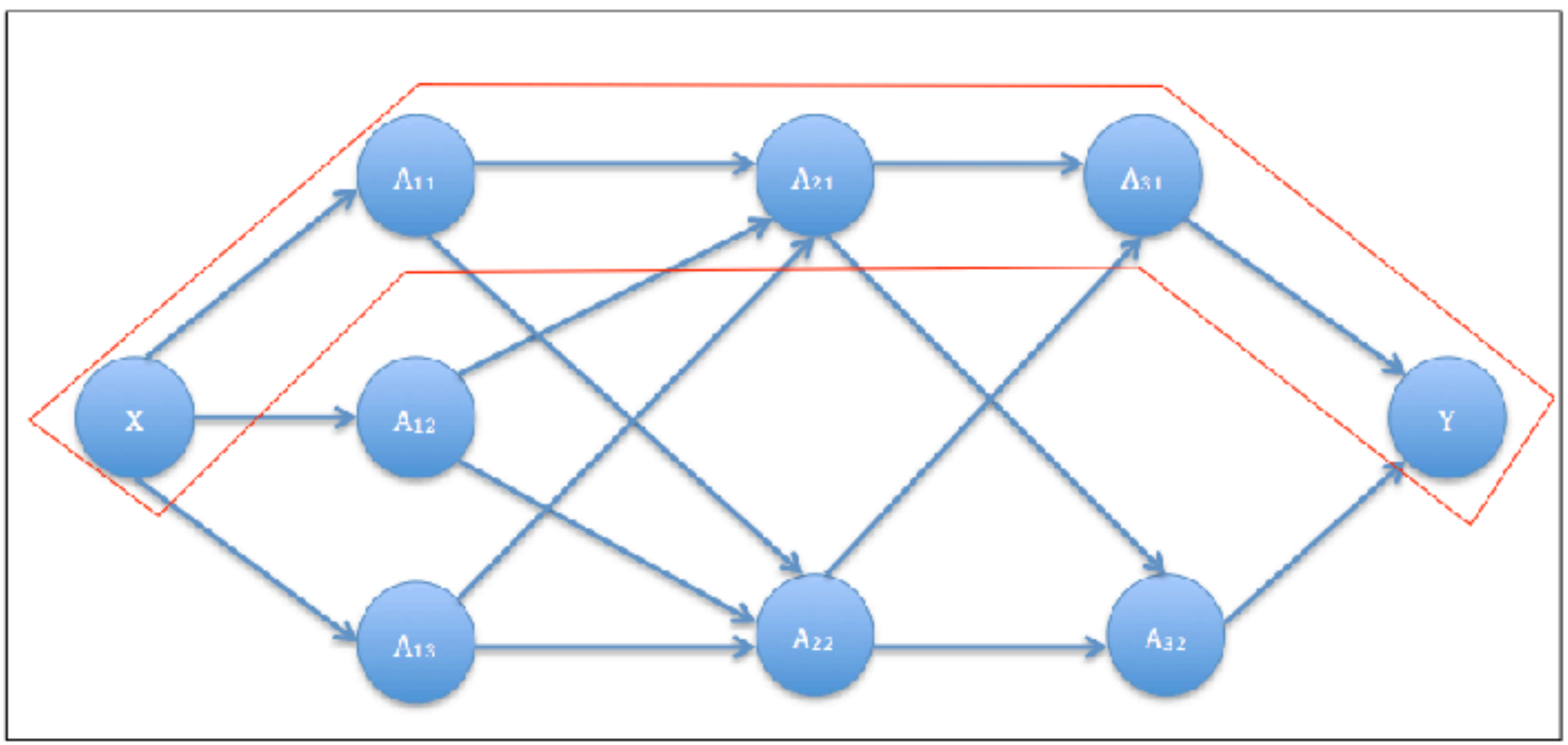
$$EC^*_{A_{ij}} = \frac{1}{|\theta_{ij}|} \sum_{z=1}^{|\theta_{ij}|} {E^z_{A_{ij}}}^* - E^*_{A^p_{ij}},$$



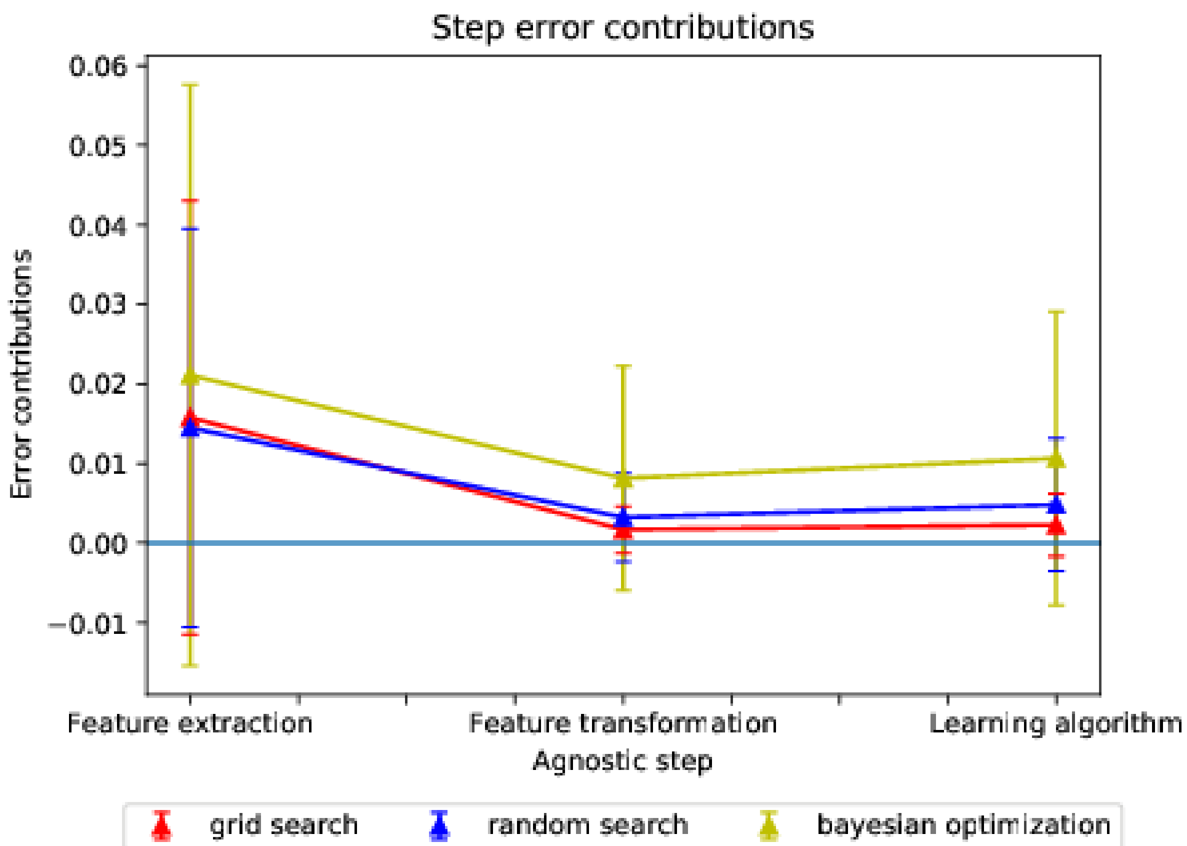
Error contribution from hyperparameters

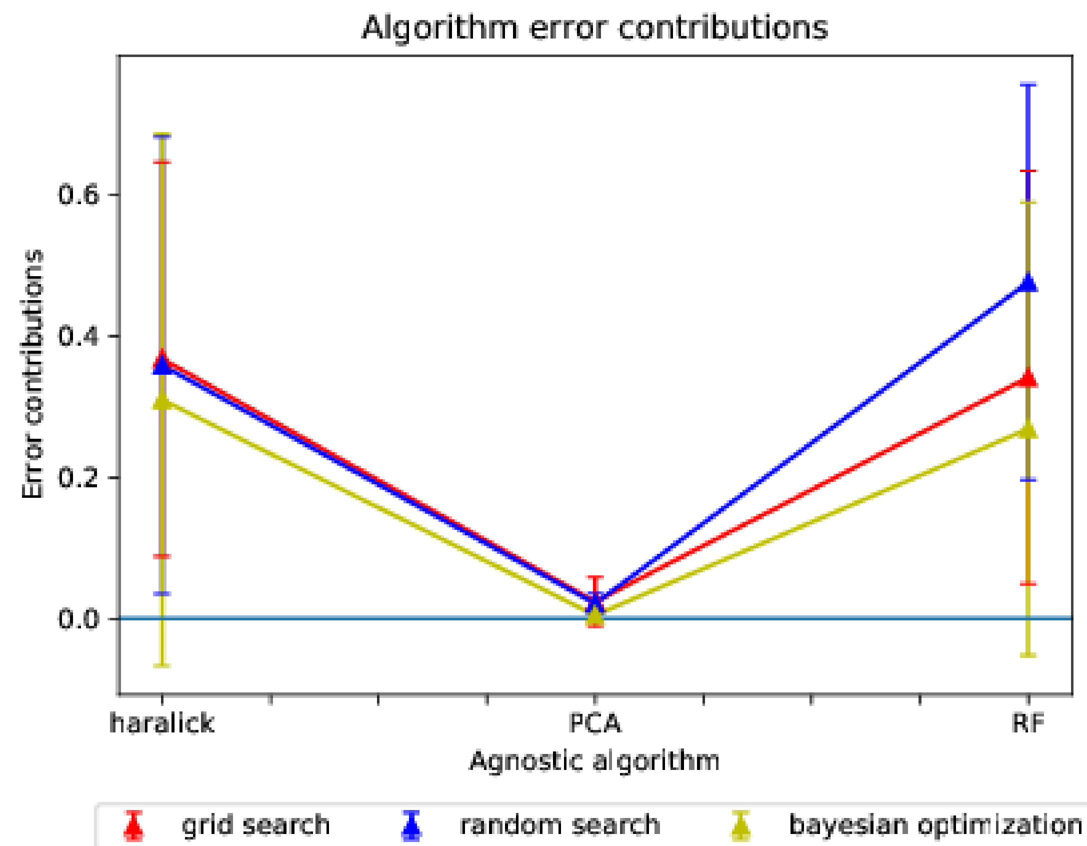
For, $i = 1, ..., n, j = 1, ..., |\theta_{ij}|$, k = number of hyper-parameters of algorithm A_{ij}. $|\theta_{ijk}|$ represents the number of configurations of θ_{ijk}, ${E^z_{\theta_{ijk}}}^*$ is the minimum error obtained with the z-th configuration of θ_{ijk} and $E^*_{A^p_{ij}}$ is the minimum error found over the path p that consists of algorithm A_{ij}.

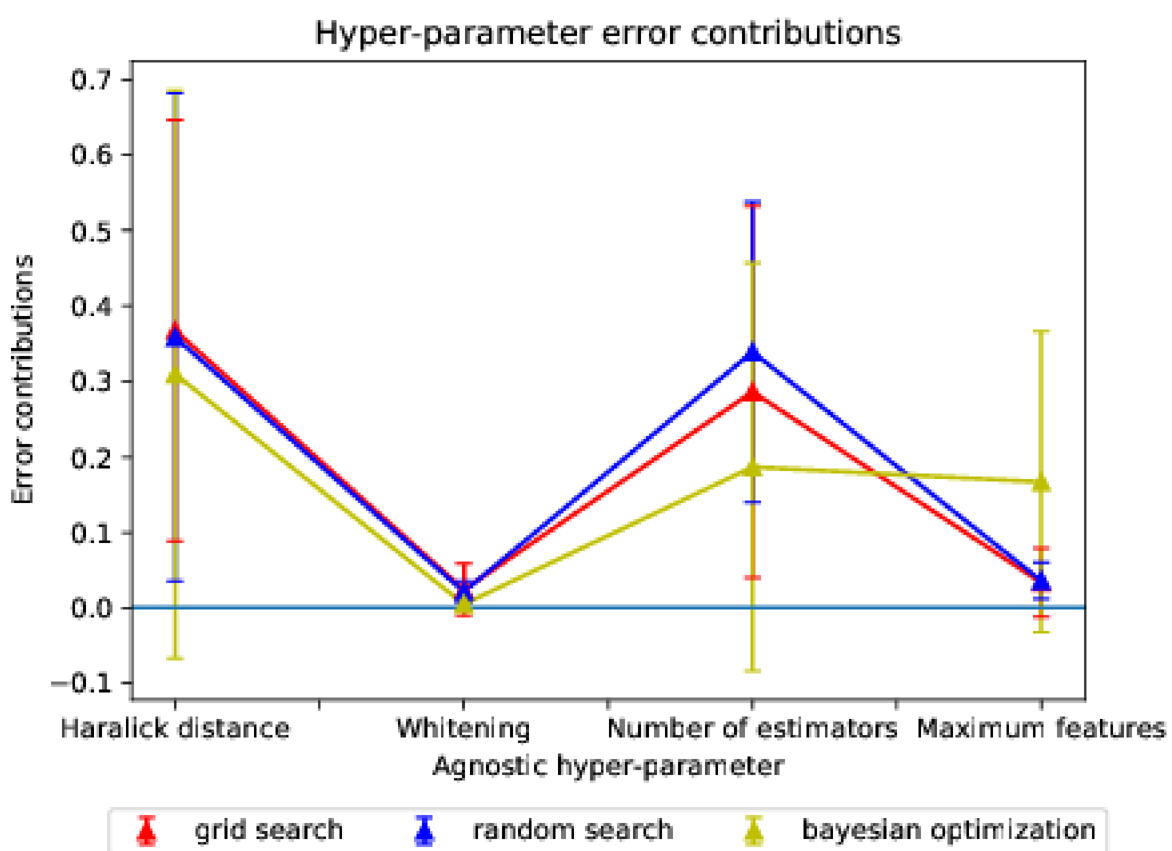
$$EC^*_{\theta_{ijk}} = \frac{1}{|\theta_{ijk}|} \sum_{z=1}^{|\theta_{ijk}|} {E^z_{\theta_{ijk}}}^* - E^*_{A^p_{ij}},$$

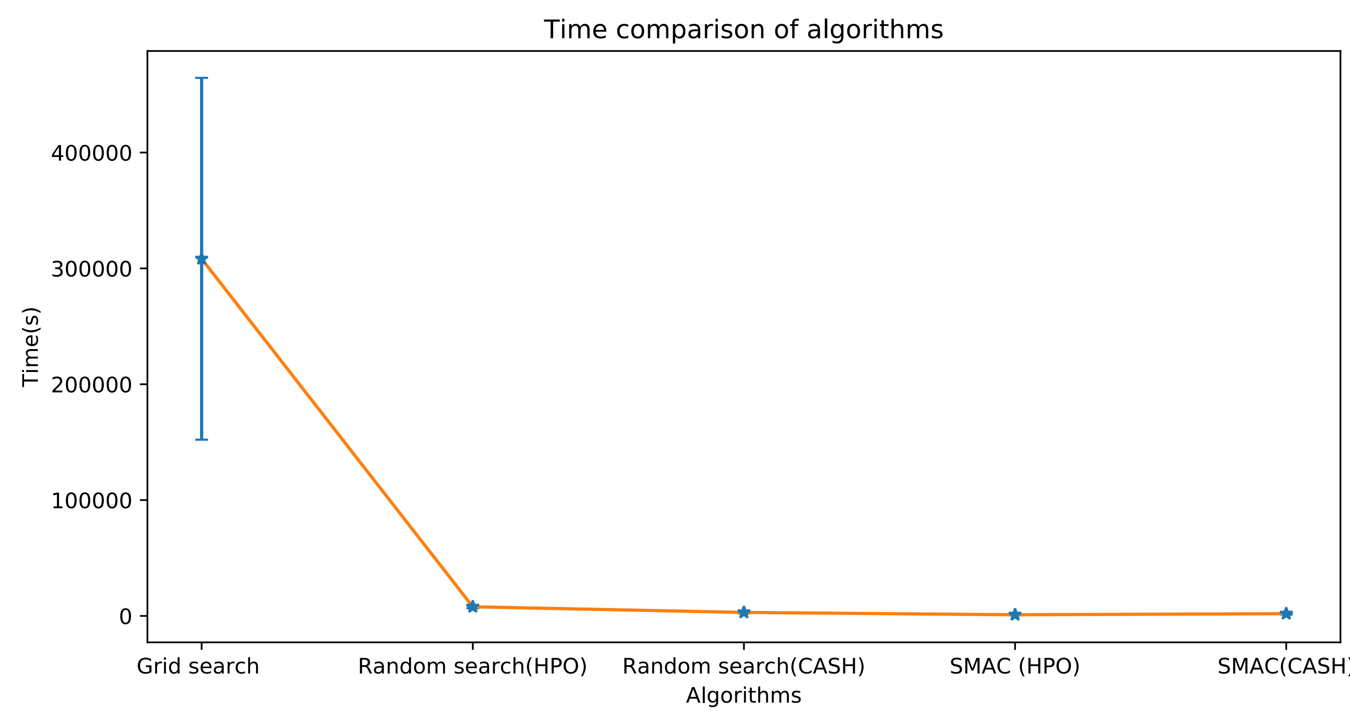


Results


Step error contributions
0.06
0.05
0.04
0.03
0.02
0.01
0.00
−0.01
Error contributions
Feature extraction
Feature transformation
Learning algorithm
Agnostic step
grid search
random search
bayesian optimization


Algorithm error contributions
0.6
0.4
0.2
0.0
Error contributions
haralick
PCA
RF
Agnostic algorithm
grid search
random search
bayesian optimization


Hyper-parameter error contributions
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
Error contributions
Haralick distance
Whitening
Number of estimators
Maximum features
Agnostic hyper-parameter
grid search
random search
bayesian optimization


Time comparison of algorithms
400000
300000
200000
100000
0
Time(s)
Grid search
Random search(HPO)
Random search(CASH)
SMAC (HPO)
SMAC(CASH)
Algorithms

Discussion

- We propose a method to quantify the contributions of components in an image classification pipeline in terms of the error.

- HPO and CASH methods (grid search, random search and Bayesian optimization) maybe used to quantify error contribution and importance of components (steps, algorithms and hyper-parameters)

- Random search is able to quantify the contributions accurately and efficiently based on the results.