Relevance propagation of variables as applied in $H \rightarrow \tau\tau$

Teresa Lenz, Mareike Meyer, Alexei Raspereza for the HIG-18-032 team

based on paper by:

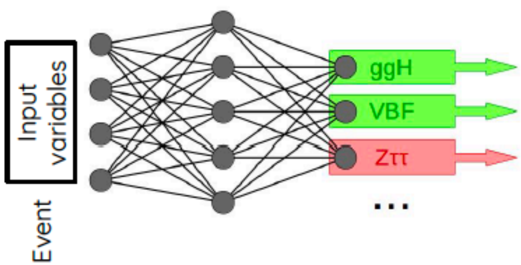
S. Wunsch, R. Friese, R. Wolf, and G. Quast, “Identifying the relevant dependencies of the neural network response on characteristics of the input space”, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782

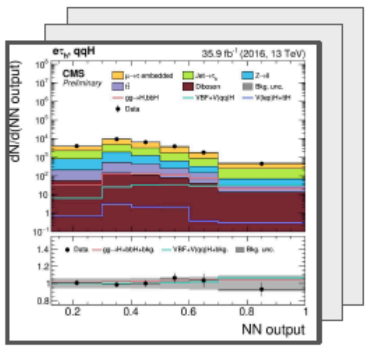
DESY CMS ML discussion, 12/07/2019

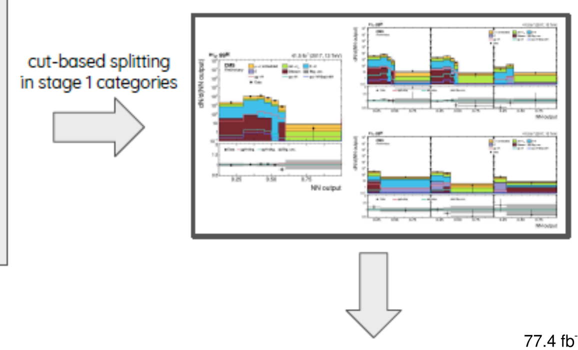
HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

Analysis strategy

- four most sensitive final states of ττ-pair studied: **eμ, eτh, μτh, τhτh**
- loose baseline selection (trigger requirements, suppression of large backgrounds)
- **multi-class NN** with **2 signal classes (ggH, qqH) & several background classes** (control regions)
- selection and validation of NN input variables based on 1D and 2D GoFs

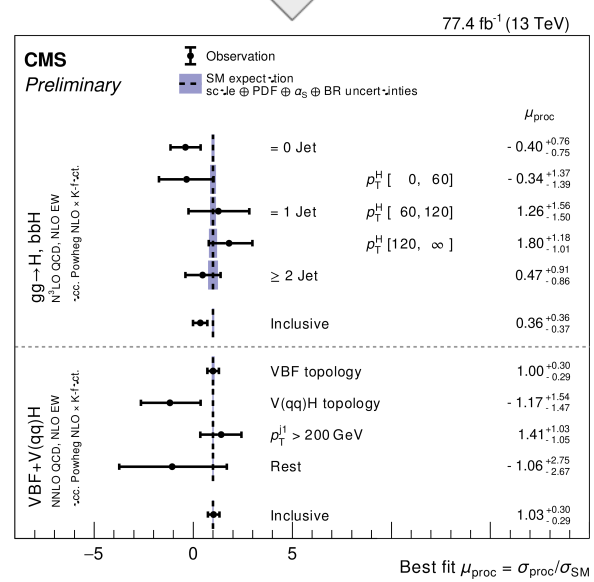






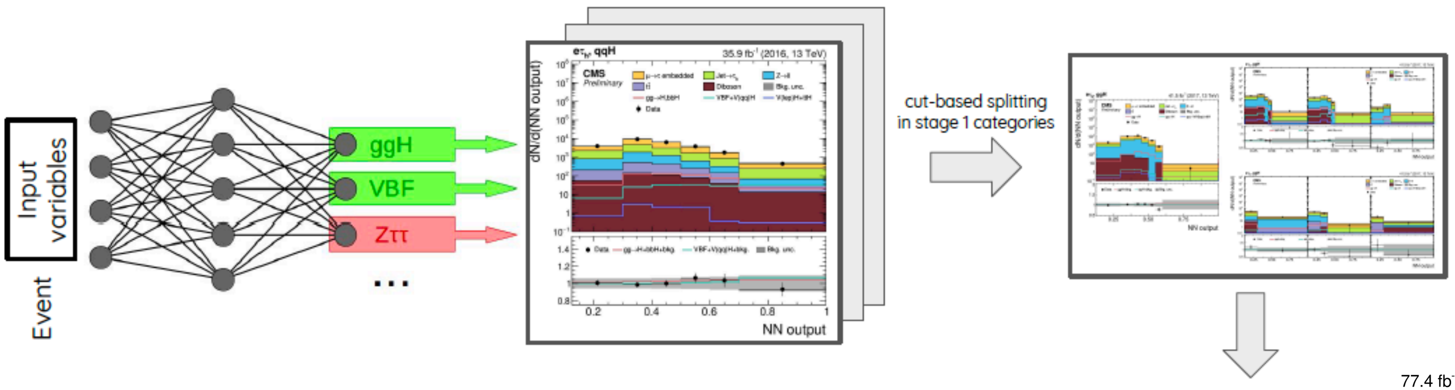
measurement of H → ττ cross sections

- inclusively
- spilt by production mode (qqH, ggH)
- in different kinematic regimes (simplified template cross sections, STXS)



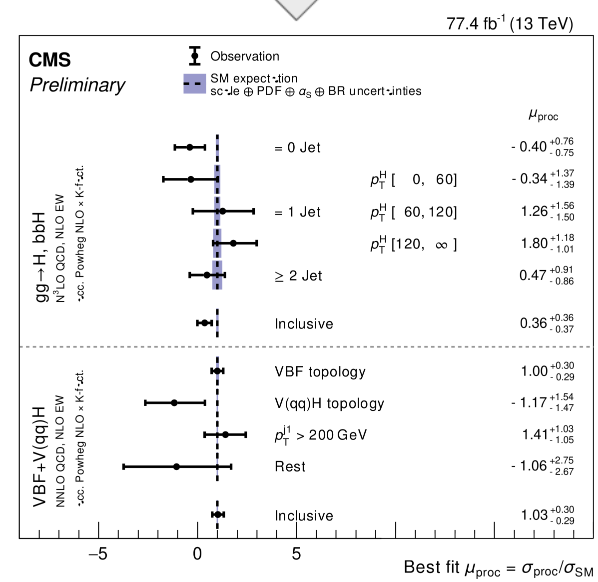
Analysis strategy

- four most sensitive final states of $\tau\tau$-pair studied: **$e\mu$, $e\tau_h$, $\mu\tau_h$, $\tau_h\tau_h$**
- loose baseline selection (trigger requirements, suppression of large backgrounds)
- **multi-class NN** with **2 signal classes (ggH, qqH) & several background classes** (control regions)
- selection & validation of NN input variables based on 1D and 2D GoFs → **relevance propagation of variables**



measurement of H → ττ cross sections

- inclusively
- spilt by production mode (qqH, ggH)
- in different kinematic regimes (simplified template cross sections, STXS)

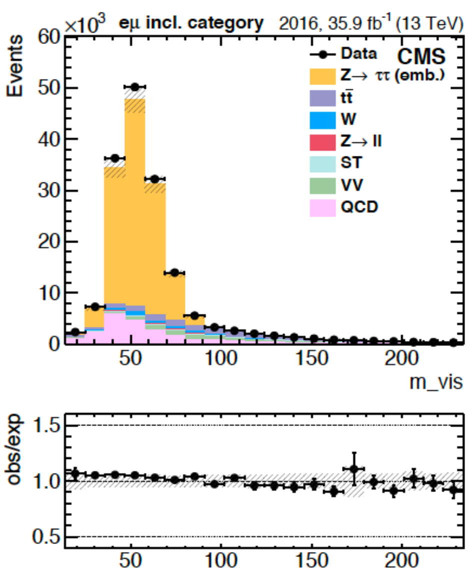


Input variable selection and validation

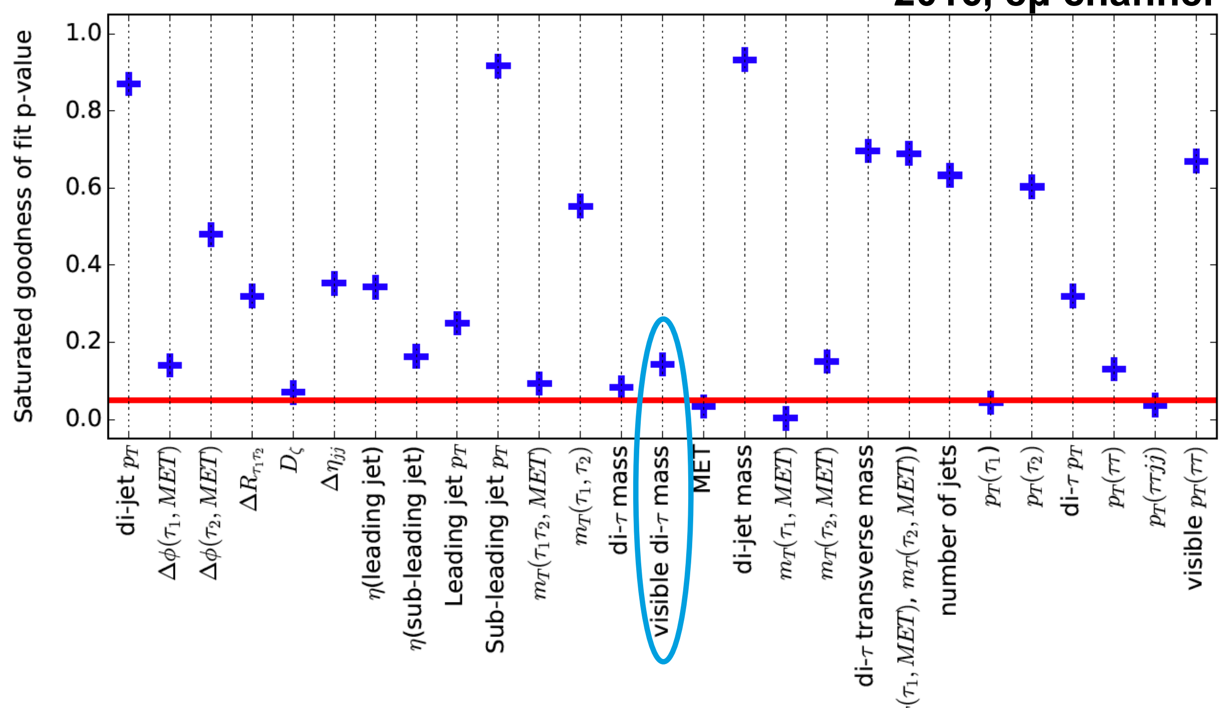
- many input variables possible
- ➡ number should be kept at manageable level, while most performant variables should be in
- input variables have to be well described by MC otherwise specifics of training set (MC) might be learned that are not present in test data
- also correlations of variables have to be well described (strength of NN to exploit correlations)

a.) **check modeling of input variables** using goodness of fit (GoF) test on 1D distributions

➡ discard variable with p-value < 5 %



2016, eμ channel

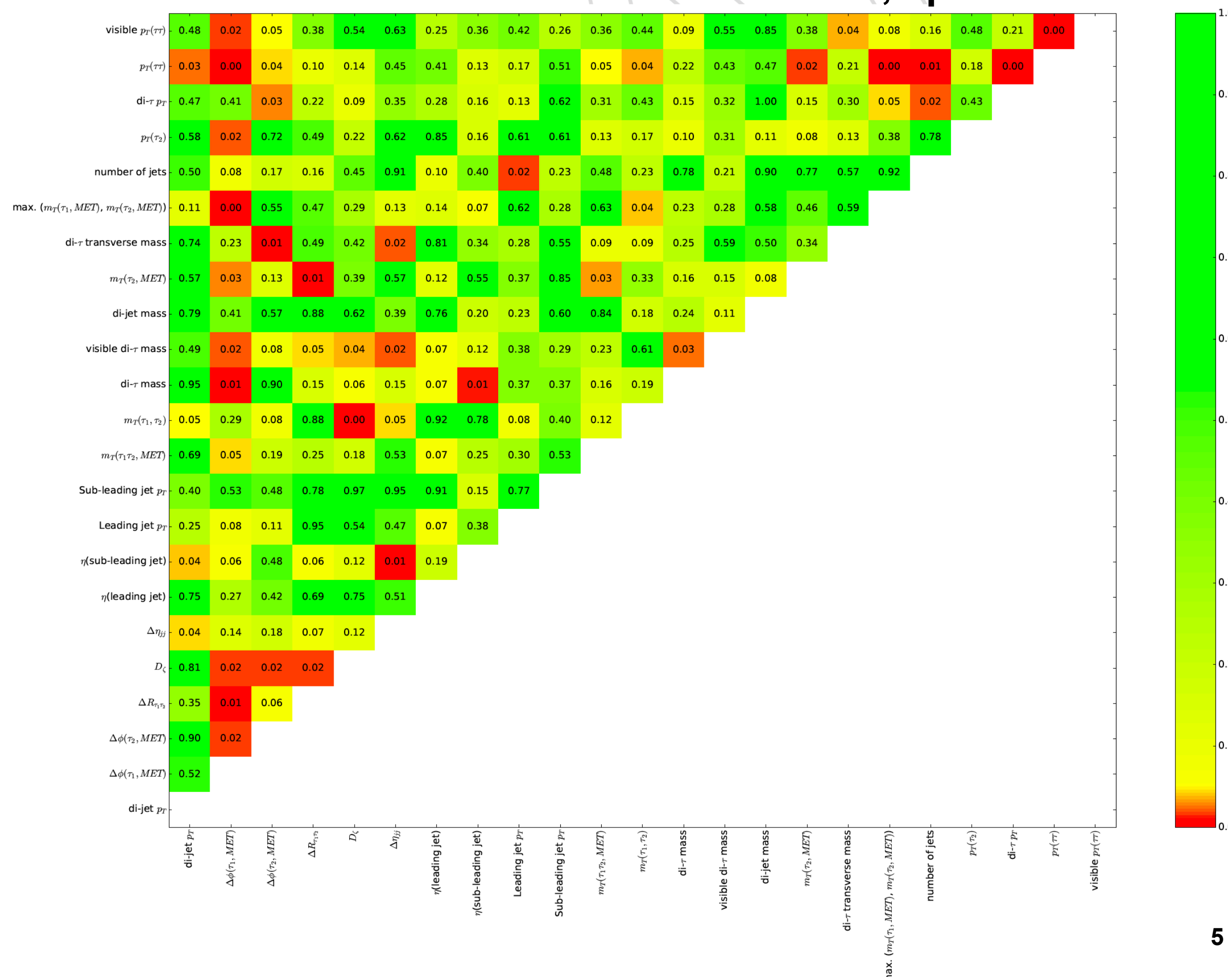


Input variable selection and validation

b.) GoF test on 2D distributions of all possible combinations of variables passing 1D GoF test to check modeling of correlations

➡study those with many low p-values

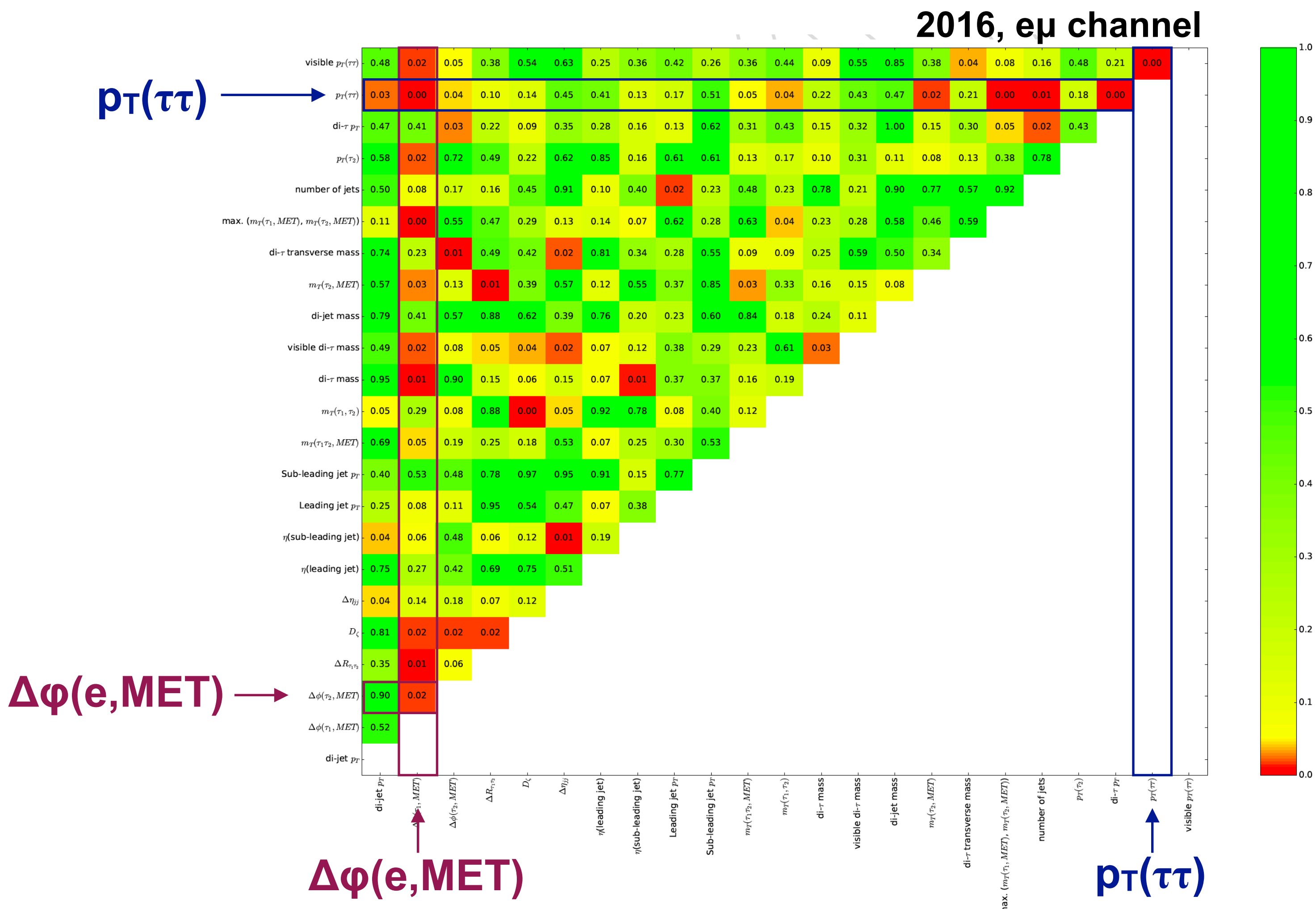
2016, eμ channel



Input variable selection and validation

b.) GoF test on 2D distributions of all possible combinations of variables passing 1D GoF test to check modeling of correlations

➡study those with many low p-values



Input variable selection and validation

c.) check relevance of variables for NN

high relevance & several badly-modeled correlations:

➡ perform further checks (e.g. modeling of variable in background classes)

➡ can we keep the variable?

low relevance & several badly-modeled correlations:

➡ discard variable

relevance of variables checked by Taylor expansion of NN output

Relevance propagation: idea

S.Wunsch, R. Friese, R.Wolf, and G. Quast, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782

- relate output space of NN to input space

➡ identify characteristics of input space that have large influence on output for a given task

- **decompose NN function into Taylor expansion in each element of the input space**
- Taylor coefficients contain information about the sensitivity of the NN response to the inputs
- dependence on phase space: mean of absolute values of Taylor coefficients evaluated for all elements of test sample

set of input variables, evaluated for element k of test sample

$$\langle t_i \rangle \equiv \frac{1}{N} \sum_{k=1}^{N} \left| t_i\left(\{x_j\}\big|_k \right) \right| \qquad i \in \mathcal{P}(\{x_j\})$$

sum over test sample of size N

i^{th} Taylor coefficient

- **first order** Taylor coefficient: **influence of single input elements**
- **second order** Taylor coefficent: **influence of pair-wise or auto-correlations**

$$T(x,y) = f(a,b) + (x-a) f_x(a,b) + (y-b) f_y(a,b) + \frac{1}{2!}\Big((x-a)^2 f_{xx}(a,b) + 2(x-a)(y-b) f_{xy}(a,b) + (y-b)^2 f_{yy}(a,b) \Big) + \cdots$$

Toy studies

S.Wunsch, R. Friese, R.Wolf, and G. Quast, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782

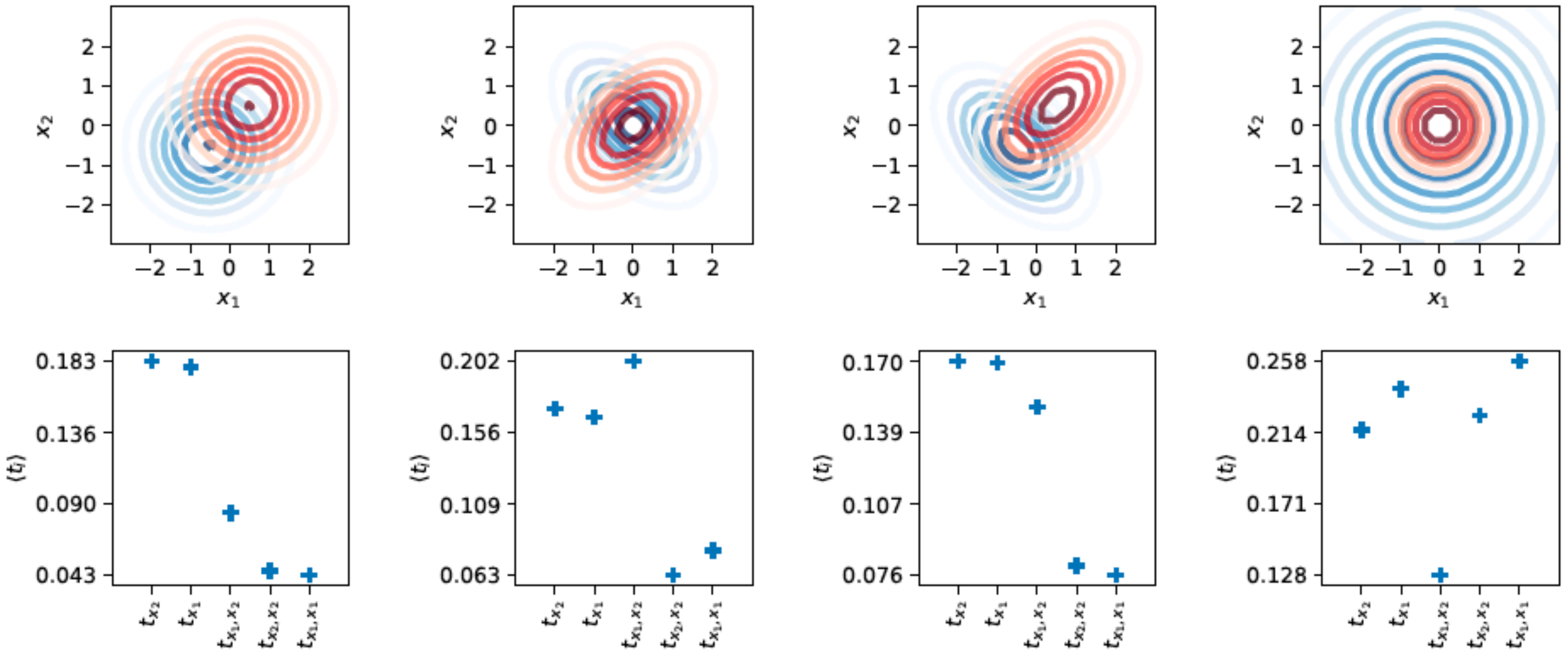
- simple task for illustration
- Keras, TensorFlow
- simple fully connected feed-forward NN
  - one hidden layer with 100 nodes
  - activation function: tanh / sigmoid
  - cross-entropy loss, Adam optimizer
- binary classification
- two inputs: x_1 and x_2 (Gaussian distributions for signal and background)

Task	Mean value				Covariance matrix	
	Signal (x_1, x_2)		Background (x_1, x_2)		Signal	Background
Fig. 1a	0.5	0.5	−0.5	−0.5	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
Fig. 1b	0	0	0	0	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}$
Fig. 1c	0.5	0.5	−0.5	−0.5	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.5 \\ -0.5 & 1 \end{pmatrix}$
Fig. 1d	0	0	0	0	$\begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}$	$\begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}$

Toy studies: results

S.Wunsch, R. Friese, R.Wolf, and G. Quast, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782

- <t_{x1}>, <t_{x2}>: influence of 1d distributions of x_1, x_2
- <$t_{x1,x2}$>: influence of the correlation of x_1 and x_2
- <$t_{x1,x1}$> , <$t_{x2,x2}$> : indicating the influence of the auto-correlation



(a) (b) (c) (d)

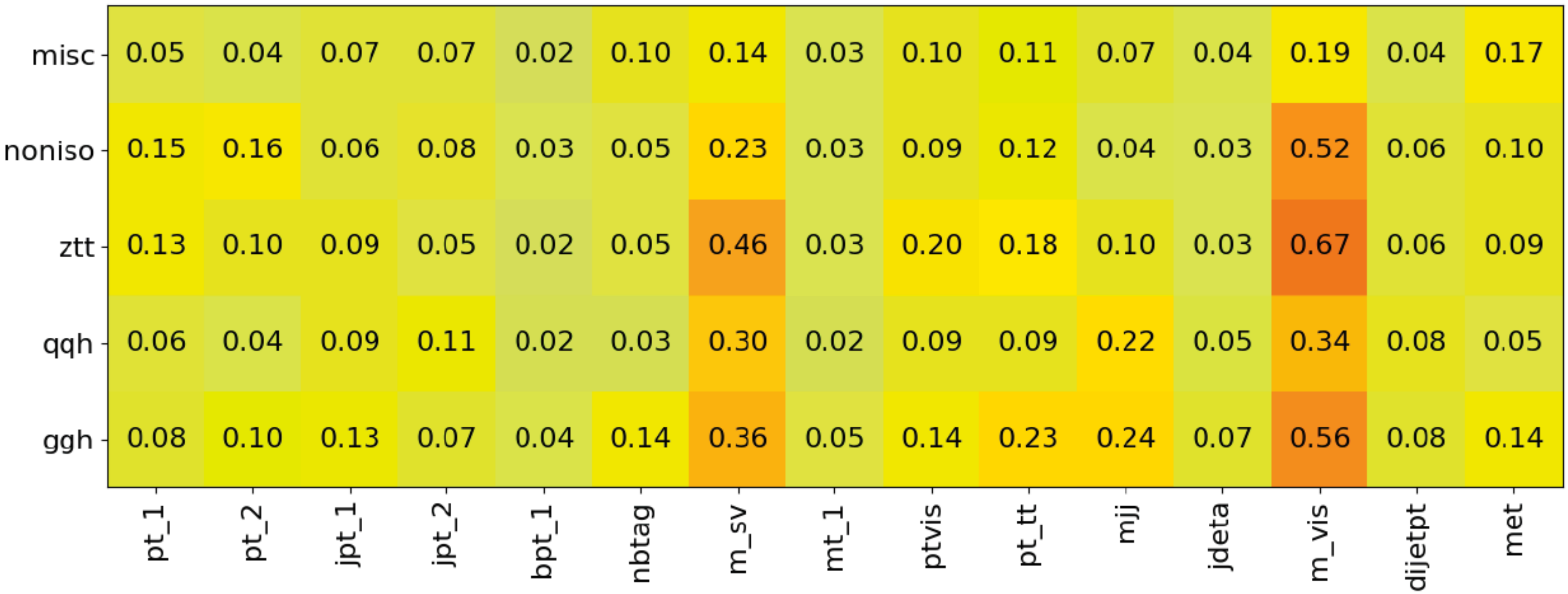
Implementation in our analysis (H $\rightarrow$ ττ)

Technical details of NN architecture:

- fully connected feed forward
- activation function: **hyperbolic tangent**, last layer: softmax
- 2 hidden layers, 200 nodes each
- dropout layer after each hidden layer (30% propability), L2 regularization (10^{-4})
- loss function: cross entropy
- optimizer: Adam (learning rate: 10^{-4})

Use Keras and Tensorflow for implementation of NN and calculation of the derivatives

Application to H → ττ: 1st order coefficients



- **we can learn which variables were important for the NN to identify the respective process**
- **note:** presents only the view of the trained NN on information in training data, training might be not optimal, there might be more information in the training data that was not picked up

Application to H $\rightarrow \tau\tau$: 2nd order coefficients

2016, $\tau_h\tau_h$ channel

ggh		
m_vis	m_vis	1.92
m_sv	m_vis	1.90
m_sv	m_sv	0.86
ptvis	m_vis	0.60
m_vis		0.59
pt_1	m_vis	0.55
m_sv	ptvis	0.54
m_sv	pt_tt	0.53
pt_tt	m_vis	0.51
pt_2	m_vis	0.48
m_vis	dijetpt	0.46
jpt_1	m_vis	0.46
mjj	m_vis	0.39
m_sv		0.37
m_vis	met	0.36
jpt_1	m_sv	0.31
m_sv	dijetpt	0.26
m_sv	mjj	0.26
mjj		0.20
jpt_2	m_vis	0.19
pt_tt		0.18
m_sv	met	0.18
pt_1	m_sv	0.18
ptvis	pt_tt	0.17
nbtag	m_vis	0.15
jpt_2	m_sv	0.15
ptvis		0.14
jdeta	m_vis	0.13
mt_1	m_vis	0.13
pt_2	m_sv	0.13
m_sv	mt_1	0.12
pt_1		0.11
jpt_1	ptvis	0.11
bpt_1	m_vis	0.11
nbtag	m_sv	0.11
met		0.11
pt_tt	mjj	0.10
jpt_1	jpt_2	0.10
ptvis	ptvis	0.10
pt_1	pt_2	0.10
pt_tt	pt_tt	0.10
mjj	mjj	0.10
pt_2		0.10
jpt_1	pt_tt	0.10
jpt_1	jpt_1	0.10
jpt_1		0.09
ptvis	mjj	0.09
ptvis	dijetpt	0.09
jpt_1	mjj	0.09
pt_tt	met	0.09

qqh		
m_sv	m_vis	1.43
m_vis	m_vis	1.21
m_sv	m_sv	0.70
m_sv	ptvis	0.48
ptvis	m_vis	0.47
m_sv	pt_tt	0.44
pt_tt	m_vis	0.42
m_vis	dijetpt	0.38
m_vis		0.36
pt_1	m_vis	0.33
pt_2	m_vis	0.28
m_sv		0.27
mjj	m_vis	0.26
m_sv	dijetpt	0.24
m_vis	met	0.23
mjj		0.21
jpt_1	m_vis	0.20
m_sv	mjj	0.16
ptvis	pt_tt	0.15
jpt_2	m_vis	0.14
jpt_1	m_sv	0.14
m_sv	met	0.13
pt_1	m_sv	0.12
mjj	mjj	0.12
jpt_2	m_sv	0.11
dijetpt		0.11
jpt_1	jpt_2	0.11
jpt_1	dijetpt	0.10
pt_2	m_sv	0.10
nbtag	m_vis	0.10
mt_1	m_vis	0.09
ptvis	ptvis	0.09
dijetpt	dijetpt	0.09
ptvis	dijetpt	0.09
mjj	dijetpt	0.09
jdeta	m_vis	0.09
jpt_2	jpt_2	0.09
pt_tt		0.08
ptvis		0.08
m_sv	mt_1	0.08
bpt_1	m_vis	0.08
pt_tt	pt_tt	0.08
jpt_2	mjj	0.08
jpt_1	mjj	0.08
pt_tt	dijetpt	0.08
jpt_2		0.08
pt_1	dijetpt	0.07
jpt_1	jpt_1	0.07
jpt_1	ptvis	0.07
jpt_1		0.07

ztt		
m_sv	m_vis	2.52
m_vis	m_vis	1.43
m_sv	m_sv	1.36
m_sv	ptvis	0.96
m_sv	pt_tt	0.90
ptvis	m_vis	0.82
pt_tt	m_vis	0.79
m_vis		0.61
m_sv		0.47
ptvis	pt_tt	0.34
jpt_1	m_sv	0.30
m_vis	dijetpt	0.29
pt_1	m_vis	0.29
jpt_1	m_vis	0.29
m_sv	dijetpt	0.26
m_sv	met	0.26
pt_2	m_vis	0.25
m_vis	met	0.22
pt_1	m_sv	0.21
pt_2	m_sv	0.20
ptvis	ptvis	0.19
ptvis		0.19
pt_tt		0.17
pt_tt	pt_tt	0.16
mjj	m_vis	0.16
m_sv	mt_1	0.15
mt_1	m_vis	0.15
m_sv	mjj	0.14
jpt_2	m_vis	0.13
jpt_2	m_sv	0.13
pt_1		0.12
jpt_1	ptvis	0.12
nbtag	m_vis	0.12
nbtag	m_sv	0.12
jpt_1	pt_tt	0.11
ptvis	met	0.11
pt_tt	met	0.11
mjj		0.10
jdeta	m_vis	0.10
ptvis	dijetpt	0.09
pt_2		0.09
bpt_1	m_sv	0.09
jpt_1		0.09
bpt_1	m_vis	0.09
pt_tt	dijetpt	0.09
m_sv	jdeta	0.08
dijetpt		0.08
pt_1	ptvis	0.08
met		0.08
pt_1	pt_tt	0.07

noniso		
m_vis	m_vis	1.01
m_vis		0.53
m_sv	m_vis	0.50
pt_1	m_vis	0.40
pt_2	m_vis	0.36
m_vis	met	0.27
m_vis	dijetpt	0.23
m_sv		0.23
jpt_1	m_vis	0.21
m_sv	m_sv	0.20
ptvis	m_vis	0.19
pt_tt	m_vis	0.18
pt_1		0.18
m_sv	ptvis	0.16
pt_2		0.15
m_sv	pt_tt	0.14
jpt_2	m_vis	0.13
nbtag	m_vis	0.13
pt_1	m_sv	0.11
jpt_1	m_sv	0.11
m_sv	met	0.11
pt_tt		0.10
met		0.10
mjj	m_vis	0.10
pt_1	pt_2	0.10
ptvis		0.10
pt_2	m_sv	0.09
bpt_1	m_vis	0.08
jpt_2		0.08
met	met	0.08
jdeta	m_vis	0.08
ptvis	pt_tt	0.07
pt_1	met	0.07
dijetpt		0.07
jpt_2	m_sv	0.07
jpt_1		0.06
jpt_2	met	0.06
jpt_1	ptvis	0.06
m_sv	dijetpt	0.06
pt_1	pt_1	0.06
pt_2	met	0.06
nbtag	met	0.06
mt_1	m_vis	0.06
jpt_1	met	0.06
pt_1	pt_tt	0.06
pt_1	dijetpt	0.05
m_sv	mjj	0.05
jpt_1	jpt_1	0.05
pt_2	dijetpt	0.05
pt_1	ptvis	0.05

misc		
m_sv	m_vis	0.60
m_vis	m_vis	0.60
m_sv	m_sv	0.25
pt_tt	m_vis	0.22
ptvis	m_vis	0.21
pt_1	m_vis	0.21
m_vis		0.20
pt_2	m_vis	0.17
jpt_1	m_vis	0.17
m_sv	ptvis	0.17
m_sv	pt_tt	0.16
m_vis	met	0.16
met		0.15
m_vis	dijetpt	0.15
m_sv		0.14
jpt_1	m_sv	0.11
mjj	m_vis	0.11
jpt_2	m_vis	0.11
m_sv	met	0.10
pt_1	m_sv	0.09
m_sv	dijetpt	0.09
pt_tt		0.09
nbtag		0.08
nbtag	m_vis	0.08
ptvis		0.08
pt_2	m_sv	0.08
jpt_2	m_sv	0.07
jpt_2		0.07
ptvis	pt_tt	0.07
m_sv	mjj	0.07
jpt_1		0.06
mt_1	m_vis	0.06
met	met	0.06
jdeta	m_vis	0.06
mjj		0.05
bpt_1	m_vis	0.05
jpt_1	ptvis	0.05
pt_tt	met	0.05
jpt_1	met	0.05
jpt_2	met	0.05
nbtag	m_sv	0.05
nbtag	met	0.05
jpt_1	dijetpt	0.05
pt_1		0.05
ptvis	met	0.05
jpt_1	jpt_2	0.05
jpt_1	jpt_1	0.05
pt_1	pt_2	0.04
pt_1	met	0.04
pt_1	pt_tt	0.04

- 50 highest ranked variables or combinations

Selected Variables

Using **1d & 2d GoFs** in combination with **1st & 2nd order Taylor coefficients**

➡ **17 to 22 input variables** depending on final state and year

$m_{\tau\tau}$ | ✓✓ ✓✓ ✓✓ ✓✓ (first ✓ of each pair: used for 2016; second ✓: used for 2017)

Variable	$e\mu$	$e\tau_h$	$\mu\tau_h$	$\tau_h\tau_h$
$m_{\tau\tau}^{SV}$	✓✓	✓✓	✓✓	✓✓
$m_{T\tau\tau}^{SV}$	✓✓	––	––	––
$p_{T\tau\tau}^{SV}$	✓✓	––	––	––
m_{vis}	✓–	✓–	✓–	✓✓
p_T^{vis}	✓✓	✓✓	✓–	✓–
$p_T^{\tau_1}$	––	––	✓–	✓✓
$p_T^{\tau_2}$	✓–	✓✓	✓✓	✓–
$\Delta R^{e\mu}$	✓✓	––	––	––
$p_T(\mathrm{jet}_1)$	✓✓	✓✓	✓✓	✓–
$\eta\ (\mathrm{jet}_1)$	✓–	––	––	––
$p_T(\mathrm{jet}_2)$	✓✓	✓✓	✓✓	✓✓
$\eta\ (\mathrm{jet}_2)$	✓–	––	––	––
m_{jj}	✓✓	✓✓	✓✓	✓✓
$\Delta\eta_{jj}$	✓✓	✓✓	✓✓	✓✓

Variable	$e\mu$	$e\tau_h$	$\mu\tau_h$	$\tau_h\tau_h$
p_T^{jj}	✓✓	✓✓	✓✓	✓✓
$p_T(\mathrm{b\ jet}_1)$	––	✓✓	✓✓	✓✓
$p_T(\mathrm{b\ jet}_2)$	––	✓✓	✓✓	✓✓
p_T^{miss}	–✓	✓✓	✓–	✓–
D_ζ	✓✓	––	––	––
m_T^{e}	––	✓✓	––	––
m_T^{μ}	✓✓	––	✓✓	––
$m_T^{e+\mu}$	✓–	––	––	––
$\max(m_T^{\mu}, m_T^{e})$	✓✓	––	––	––
$m_T^{\tau_h}$	––	✓✓	✓–	✓✓
$p_T^{\tau\tau+miss}$	✓✓	✓✓	✓✓	✓✓
$p_T^{\tau\tau jj+miss}$	–✓	––	––	––
$N_{b\ jet}$	––	✓✓	✓✓	✓✓
N_{jet}	✓✓	✓✓	✓✓	–✓

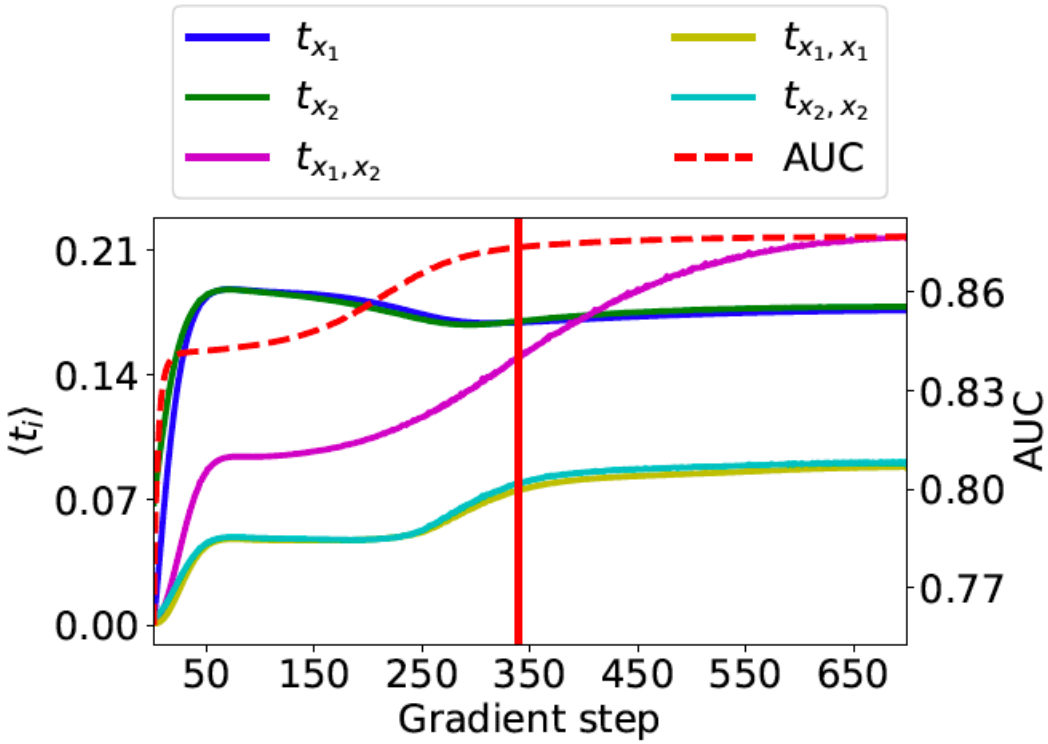
currently large number of variables used, different between channels and years

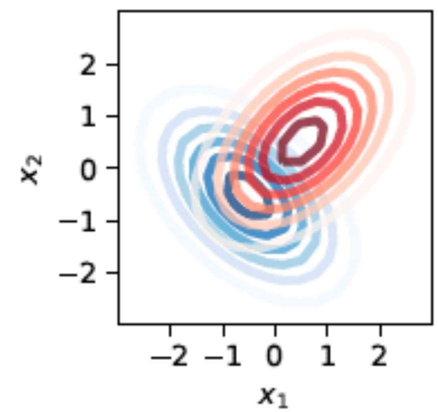
➡ studies for harmonization and reduction of variables on-going

- Taylor expansion important tool to identify the relevant variables

Further ideas for application: analyze the learning process

S.Wunsch, R. Friese, R.Wolf, and G. Quast, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782





- evaluate Taylor coefficients at each minimization step during training

➡monitor training progress, can help to interpret features and NN sensitivity to them

- **but** different NN configurations can look different, no proof of influence, training may not be optimal

Summary

- presented relevance propagation as applied in $H \rightarrow \tau\tau$
- Taylor expansion can be used to get an understanding of the learned properties and to monitor the training process
- used in $H \rightarrow \tau\tau$ to check if mis-modeled variables or correlations of variables are important for the trained NN and therefore could have an influence on the final result
- **more information in :**
  - S.Wunsch, R. Friese, R.Wolf, and G. Quast, “Identifying the relevant dependencies of the neural network response on characteristics of the input space”, Computing and Software for Big Science 2 (Sep, 2018) 5, doi:10.1007/s41781-018-0012-, arXiv:1803.08782
  - CMS AN - 2018/256

Thank you for your attention!