Canonical Correlation Forests

Tom Rainforth, Frank Wood

{twgr,fwood}@robots.ox.ac.uk, www.robots.ox.ac.uk/~twgr

UNIVERSITY OF OXFORD

Replacement for Random Forest

- Canonical correlation forests (CCFs) [1] are a new decision tree ensemble learning method for classification
- CCFs are composed of canonical correlation trees (CCTs) which use hyperplane splits based on canonical correlation components
- CCFs require no parameter tuning
- CCFs outperform both the state-of-art tree ensemble methods random forest (RF) [2] and rotation forest [3] significantly

Trees and Forests Review

- Decision trees hierarchically divide the input space and assign local models to the leafs
- Classical decision tree training algorithms greedily search the possible space of axis-aligned unique splits
- Combining individual trees to form a forest improves performance

Canonical Correlation Analysis

- CCA [4] is used to give pairs of projections that maximise the correlation between the features *X* and the class labels *Y*
  $$\{\Phi, \Omega\} = \mathrm{CCA}\left(X, Y\right) = \mathrm{argmax}_{a,b}\left(\mathrm{corr}\left(Xa, Yb\right)\right)$$
- $\min\left(\mathrm{rank}\left(X\right), \mathrm{rank}\left(Y\right)\right)$ pairs are produced by adding the constraint that new components are uncorrelated with previous components
- e.g.

$$\left\{\begin{bmatrix}-2.11 & -2.49\\ 0.52 & 0.93\end{bmatrix}, \begin{bmatrix}0 & 0\\ -1.34 & -2.49\\ -2.28 & -0.38\end{bmatrix}\right\} = \mathrm{CCA}\left(\begin{bmatrix}1 & 0.5\\ 2 & 2\\ 3 & 4.5\\ 4 & 8\\ 5 & 12.5\end{bmatrix}, \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\\ 0 & 0 & 1\\ 0 & 0 & 1\end{bmatrix}\right)$$

- Unaffected by affine transformation or rotation

CCF Training

Algorithm 1 Canonical correlation forest training algorithm

Inputs: $X^r \in \mathbb{R}^{N\times D^r},\ X^c \in \mathcal{S}^{N\times D^c},\ \mathcal{Y} \in \mathbb{I}^{N\times K}, L \in \mathbb{Z}^+,\ \lambda \in \mathbb{Z}^+$ ▷ $\mathcal{S}$ represents non-ordinal space
Convert X^c to 1-of-K encoding $X^b \in \mathbb{I}^{N\times D^b}$, $\quad X = \{X^r, X^b\}$
$\mu_{(d)} = \sum_{n=1}^{N} X_{(n,d)}/N, \quad \sigma_{(d)} = \sqrt{\sum_{n=1}^{N} {X_{(n,d)}}^2 - {\mu_{(d)}}^2}/\sqrt{(N-1)} \quad \forall d$
$X_{(:,d)} \leftarrow \left(X_{(:,d)} - \mu_{(d)}\right)/\sigma_{(d)} \quad \forall d,$ set missing values in X to 0
if $\lambda < (D^r + D^c)$ **then** $b =$ true **else** $b =$ false **end if**
for $i = 1 : L$ **do**
  if b **then** $\{X', \mathcal{Y}'\} \leftarrow \{X, \mathcal{Y}\}$ **else** $\{X', \mathcal{Y}'\} \leftarrow$ sample with replacement N rows from $\{X, \mathcal{Y}\}$ **end if**
  $[\cdot, \Psi, \Theta] = \textsc{GrowTree}(X', \mathcal{Y}', \{1, \dots, D^r + D^c\}, \lambda, b)$
  $t_i = \{\Psi, \Theta\}$
end for
return $T = \{t_i\}_{i=1\dots L}$

References

[1] Tom Rainforth, and Frank Wood. Canonical Correlation Forests, 2015. Preprint and code available at http://robots.ox.ac.uk/~twgr

[2] Leo Breiman. Random forests. Machine learning, 45(1):5–32, 2001.

[3] Juan José Rodriguez, Ludmila I Kuncheva, and Carlos J Alonso. Rotation forest: A new classifier ensemble method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(10):1619–1630, 2006.

[4] Magnus Borga. Canonical correlation: a tutorial. *On line tutorial* http://people.imt.liu.se/magnus/cca, 4, 2001.

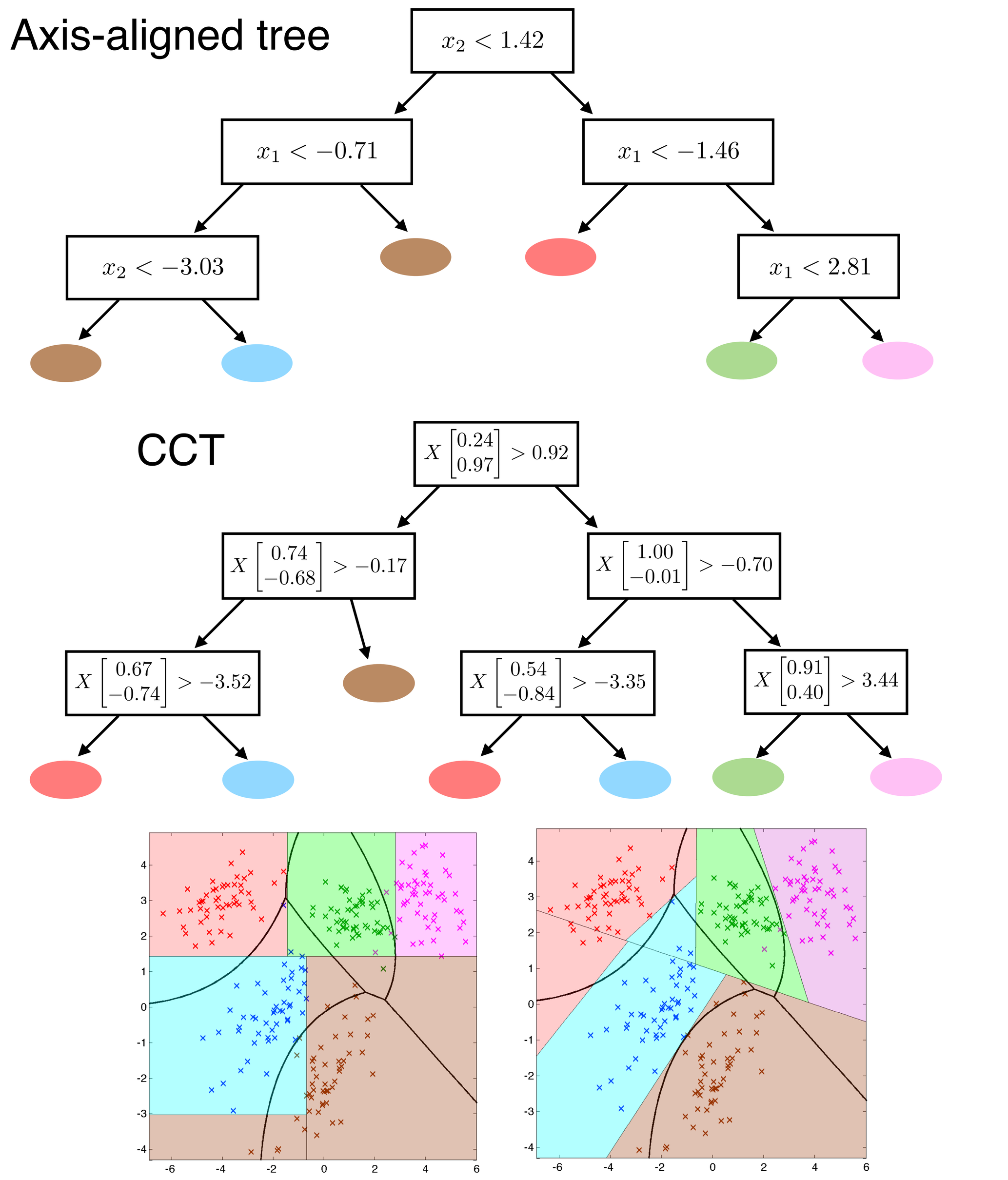
[5] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[6] Tin Kam Ho. The random subspace method for constructing decision forests. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):832–844, 1998.

[7] M. Lichman. UCI machine learning repository, 2013. https://archive.ics.uci.edu/ml/datasets.html

Coordinate Free Splitting

- CCTs allow non axis aligned partitions, $X\phi > s$ for features *X*, projection vector ϕ and split point s



Corresponding input space partionings with Bayes optimal decision surface in bold

Projection Bootstrap - A New Idea

- Instead of using tree bagging [5], perform CCA on a local bootstrap sample of the data
- Use all data points to select the split point and best projection in the projected space

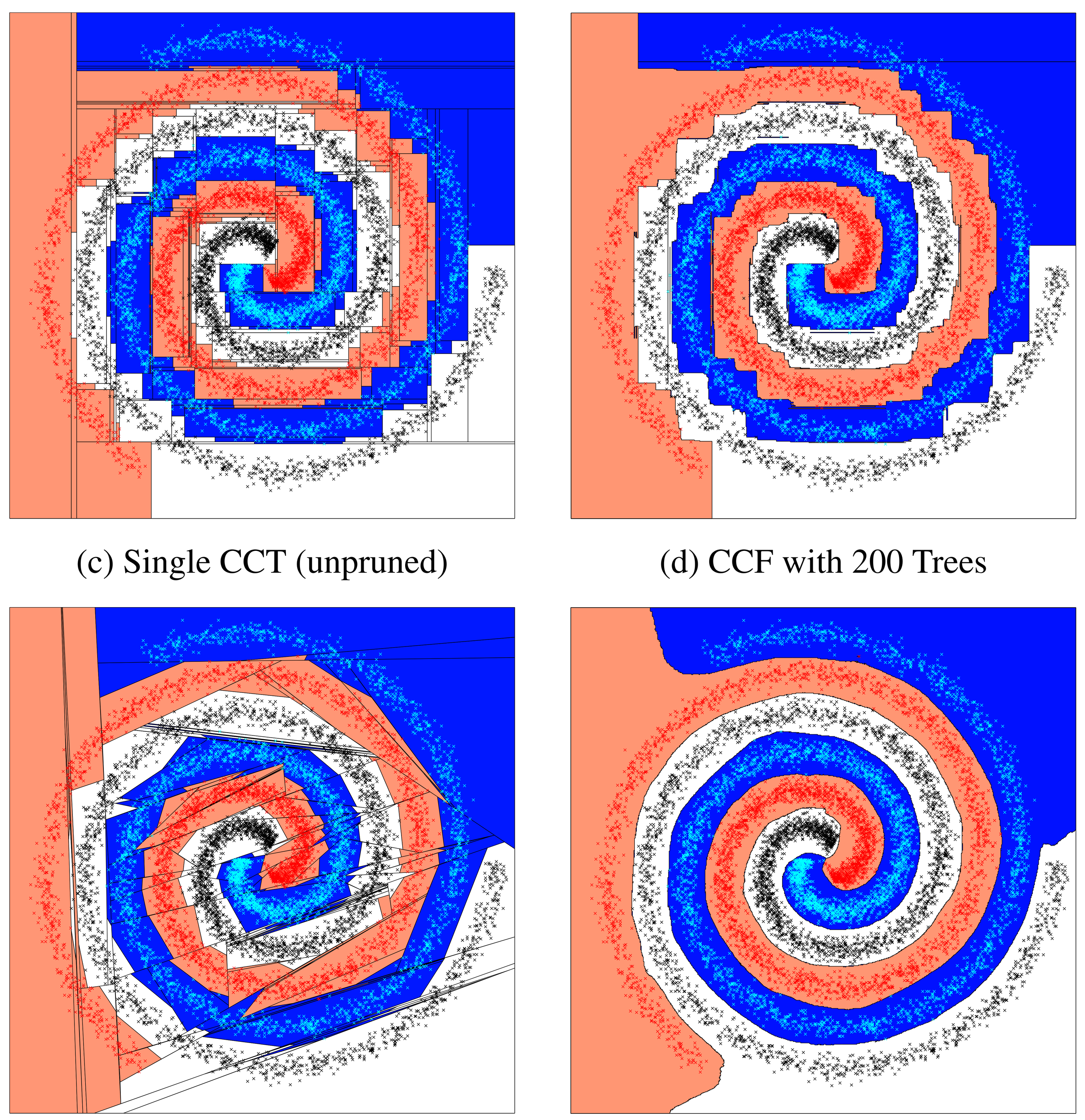
Growing CCTs

- Randomly subsample features [6]
- On local bootstrap sample, carry out CCA between features and class labels
- Calculate best split in projected space using original data (projection bootstrapping)
- If no improvement is possible assign as leaf

Algorithm 2 GrowTree

Inputs $X^j \in \mathbb{R}^{N^j\times D^r+D^b},\ \mathcal{Y}^j \in \mathbb{I}^{N^j\times K},\ \mathcal{D}^j \subseteq \{1, \dots, D^r + D^c\},\ \lambda \in \mathbb{Z}^+, b \in \{\text{true}, \text{false}\}$
Set current node index j to an unique node identifier (0 for root node)
Sample $\delta \subseteq \mathcal{D}^j$ by taking $\min\left(\lambda, \left|\mathcal{D}^j\right|\right)$ samples without replacement from $\mathcal{D}^j$
While δ contains features without variation, eliminate these from $\mathcal{D}^j$ and δ and resample
$\gamma = \delta$ mapped to the column indices of X^j in accordance with the 1-of-K encoding of X^c
if b **then** $\{X', \mathcal{Y}'\} \leftarrow$ sample with replacement N^j rows from $\{X^j_{(:,\gamma)}, \mathcal{Y}^j\}$
else $\{X', \mathcal{Y}'\} \leftarrow \{X, \mathcal{Y}\}$ **end if**
if all rows in X' or $\mathcal{Y}'$ are identical **then**
  if all rows in $X^j_{(:,\gamma)}$ or $\mathcal{Y}^j$ are identical **then return** $\left[j, \emptyset, \textsc{label}(\mathcal{Y}^j)\right]$ **end if**
  $\{X', \mathcal{Y}'\} \leftarrow \{X^j_{(:,\gamma)}, \mathcal{Y}^j\}$
end if
if X' contains only two unique rows **then**
  $\mathcal{X}' = \textsc{UniqueRows}(X')$
  $\phi_{j\,(\gamma)} \leftarrow \mathcal{X}'_{(2,:)} - \mathcal{X}'_{(1,:)}, \quad \phi_{j\,(:\setminus\gamma)} \leftarrow 0$
else
  $[A, \cdot] = \mathrm{CCA}(X', \mathcal{Y}')$
  $R_{(\gamma,:)} \leftarrow A, \quad R_{(:\setminus\gamma,:)} \leftarrow 0$
  $U = X^j R$
  $[\xi, s_j, \text{gain}] = \textsc{FindBestSplit}(U)$
  if gain ≤ 0 **then return** $\left[j, \emptyset, \textsc{label}(\mathcal{Y}^j)\right]$ **end if**
  $\phi_j = R_{(:,\xi)}$
end if
$\tau_l = \left\{n \in \{1, \dots, N^j\} : X^j_{(n,:)}\phi_j \leq s_j\right\}, \quad \tau_r = \{1, \dots, N^j\} \setminus \tau_l$
$\left[\chi_{(j,1)}, \Psi_l, \Theta_l\right] = \textsc{GrowTree}(X^j_{(\tau_l,:)}, \mathcal{Y}^j_{(\tau_l,:)}, \mathcal{D}^j, \lambda, b)$
$\left[\chi_{(j,2)}, \Psi_r, \Theta_r\right] = \textsc{GrowTree}(X^j_{(\tau_r,:)}, \mathcal{Y}^j_{(\tau_r,:)}, \mathcal{D}^j, \lambda, b)$
$\psi_j = \left\{\chi_{(j,1)}, \chi_{(j,2)}, \phi_j, s_j\right\}$
return $[j, \{\psi_j \cup \Psi_l \cup \Psi_r\}, \{\Theta_l \cup \Theta_r\}]$

CCFs have Smoother Decision Surfaces



(a) Single CART (unpruned) (b) RF with 200 Trees (c) Single CCT (unpruned) (d) CCF with 200 Trees

Experiments

- 37 datasets [7]
- Compare against random forest, rotation forest and CCF-Bag which uses bagging instead of the projection bootstrap

	CCF	CCF-Bag	RF	Rotation Forest
CCF	-	2	2	7
CCF-Bag	**18**	-	1	14
RF	**26**	27	-	25
Rotation Forest	**13**	10	4	-

Number of significant victories column vs row at 1% level of Wilcoxon signed ranked test

Conclusions

- CCFs outperforms all other methods significantly
- Computationally less expensive than rotation forest and similar to RF
- Unaffected by correlation between features
  - RF poor at dealing with such correlations
  - Rotation forests can only adapt to global correlations
- New benchmark for out-of-box tree ensemble classification
- Concepts introduced apply to forest regression models

Results

Data set	K	N	D^c	D^r	CCF	CCF-Bag		RF		Rotation Forest	
Balance scale	3	625	0	4	91.06 ± 3.74	91.26 ± 3.48		83.81 ± 4.23	•	**92.71 ± 3.43**	○
Banknote	2	1372	0	4	**100.00 ± 0.00**	**100.00 ± 0.00**		99.28 ± 0.75	•	**100.00 ± 0.00**	
Breast tissue	6	106	0	9	**71.58 ± 11.79**	71.09 ± 12.38		69.03 ± 12.99	•	**71.58 ± 12.70**	
Climate crashes	2	360	0	18	**94.17 ± 4.04**	93.56 ± 4.09	•	92.87 ± 4.18	•	94.04 ± 3.89	
Fertility	2	100	0	9	86.73 ± 9.38	86.47 ± 9.70		86.40 ± 9.78		**87.67 ± 8.78**	
Heart-SPECT	2	267	0	22	**82.84 ± 7.13**	81.98 ± 6.71	•	81.38 ± 7.27	•	82.49 ± 7.29	
Heart-SPECTF	2	267	0	44	81.46 ± 7.34	**81.98 ± 6.89**		81.01 ± 6.71		81.23 ± 7.29	
Hill valley	2	1212	0	100	**100.00 ± 0.00**	**100.00 ± 0.00**		61.02 ± 4.34	•	93.74 ± 2.66	•
Hill valley noisy	2	1212	0	100	**94.99 ± 1.85**	94.38 ± 2.02	•	57.98 ± 4.41	•	88.75 ± 2.84	•
ILPD	2	640	0	10	71.97 ± 5.05	**72.03 ± 5.21**		70.34 ± 4.95	•	70.98 ± 5.20	
Ionosphere	2	351	0	33	**95.12 ± 3.63**	94.38 ± 3.78	•	93.56 ± 3.89	•	94.30 ± 3.51	•
Iris	3	150	0	4	97.56 ± 3.89	**97.69 ± 3.78**		94.93 ± 5.39	•	95.82 ± 5.10	•
Landsat satellite	2	6435	0	36	91.76 ± 1.08	91.30 ± 1.05	•	91.84 ± 1.01		**92.19 ± 0.99**	○
Letter	26	20000	0	16	**97.75 ± 0.33**	97.42 ± 0.36	•	96.64 ± 0.38	•	97.52 ± 0.32	•
Libras	15	360	0	90	89.70 ± 4.75	88.65 ± 5.23	•	81.30 ± 5.97	•	**90.28 ± 4.79**	
MAGIC	2	19020	0	10	**88.42 ± 0.72**	88.29 ± 0.72	•	88.15 ± 0.74	•	87.36 ± 0.72	•
Nursery	5	12960	0	8	99.96 ± 0.07	99.91 ± 0.11	•	99.67 ± 0.19	•	**99.96 ± 0.06**	
ORL	40	400	0	10304	**97.82 ± 2.25**	97.38 ± 2.75		97.55 ± 2.52		NaN ± NaN	
Optical digits	10	5620	0	64	**98.71 ± 0.45**	98.56 ± 0.46	•	98.35 ± 0.49	•	98.69 ± 0.40	
Parkinsons	2	195	0	22	**93.90 ± 5.43**	92.27 ± 6.01	•	90.97 ± 6.04	•	92.39 ± 5.44	•
Pen digits	10	10992	0	16	**99.60 ± 0.19**	99.54 ± 0.21	•	99.17 ± 0.29	•	99.51 ± 0.23	•
Polya	2	9255	0	169	78.82 ± 1.31	78.72 ± 1.28		78.72 ± 1.36		**79.79 ± 1.32**	○
Seeds	3	210	0	7	**95.21 ± 4.73**	94.57 ± 5.18	•	93.62 ± 5.23	•	95.14 ± 4.53	
Skin seg	2	245057	0	3	**99.97 ± 0.01**	99.97 ± 0.01	•	99.96 ± 0.01	•	99.96 ± 0.01	•
Soybean	19	683	13	22	**94.58 ± 2.94**	94.18 ± 3.14	•	94.44 ± 3.08		94.40 ± 2.92	
Spirals	3	10000	0	2	**99.73 ± 0.16**	**99.73 ± 0.16**		98.78 ± 0.33	•	98.98 ± 0.34	•
Splice	3	3190	60	0	**96.90 ± 0.97**	96.71 ± 1.12	•	96.88 ± 0.93		95.74 ± 1.18	•
Vehicle	4	846	0	18	**82.69 ± 3.93**	82.68 ± 4.08		74.74 ± 4.64	•	79.09 ± 4.39	•
Vowel-c	11	990	2	10	**99.06 ± 0.95**	98.72 ± 1.08	•	97.35 ± 1.72	•	99.00 ± 0.95	
Vowel-n	11	990	0	10	98.01 ± 1.32	97.18 ± 1.64	•	96.75 ± 1.77	•	**98.52 ± 1.23**	○
Waveform (1)	3	5000	0	21	86.42 ± 1.58	**86.52 ± 1.49**		85.04 ± 1.63	•	86.44 ± 1.56	
Waveform (2)	3	5000	0	40	86.64 ± 1.59	**86.69 ± 1.66**		85.31 ± 1.66	•	86.64 ± 1.64	
Wholesale-c	2	440	1	7	91.48 ± 3.80	91.61 ± 3.99		**91.92 ± 4.09**		91.44 ± 4.12	
Wholesale-r	3	440	0	7	69.44 ± 6.18	71.03 ± 6.27	○	71.05 ± 6.19	○	**71.82 ± 6.19**	○
Wisconsin cancer	2	699	0	9	96.71 ± 2.10	96.79 ± 1.96		96.87 ± 1.98		**97.19 ± 1.82**	○
Yeast	2	1484	0	8	61.85 ± 4.04	62.72 ± 3.80	○	62.28 ± 4.07	○	**62.75 ± 4.10**	○
Zoo	7	101	0	16	**96.73 ± 5.73**	96.27 ± 6.19		95.20 ± 6.42	•	94.33 ± 6.70	•

Above: % test accuracy 15, 10-fold cross validations, best method in bold, •/○ indicate CCF significantly better / worse at 1% level

Below: corresponding box plots