**ΥΠΟΛΟΓΙΣΤΙΚΗ ΝΟΗΜΟΣΥΝΗ**

Διδάσκοντες: Σ. Λυκοθανάσης, Δ. Κουτσομητρόπουλος

Ακαδημαϊκό Έτος 2022-2023

**Εργαστηριακή Άσκηση**

**Μέρος Α’**

Σούρλας Ζήσης

Αρ. Μητρώου: 1072477

sourlas.zisis@upnet.gr

**Α1. Προεπεξεργασία και Προετοιμασία δεδομένων**

1. Για την προεπεξεργασία των δεδομένων χρησιμοποιήθηκε η βιβλιοθήκη **sklearn**.

**Μετατροπή κατηγορικών σε αριθμητικές τιμές**

Για να μετατρέψουμε τις κατηγορικές τιμές της εισόδου σε αριθμητικές χρησιμοποιήθηκε ο **OrdinalEncoder** της **sklearn** ο οποίος κωδικοποιεί τις κατηγορικές τιμές σε ακέραιες τιμές στο διάστημα [0, πλήθος\_κατηγοριών-1]. Για την κωδικοποίηση των κλάσεων χρησιμοποιήθηκε ο **LabelEncoder** της **sklearn** ο οποίος κωδικοποιεί τις κλάσεις σε ακέραιες τιμές στο διάστημα [0, πλήθος\_κλάσεων-1] και προτείνεται από το documentation της scikit learn για κωδικοποίηση κλάσεων.

**Κεντράρισμα (Centering)**

Το κεντράρισμα των δεδομένων είναι μια μεθοδολογία που χρησιμοποιείται πολύ συχνά στην προπεξεργασία δεδομένων. Αφαιρώντας το μέσο όρο από τα δεδομένα δημιουργούμε δεδομένα με μηδενική μέση τιμή. Αυτό σημαίνει ότι κατά τη διαδικασία μάθησης μειώνεται η επίδραση των ακραίων τιμών που μπορεί να οδηγήσουν σε καθυστέρηση στη σύγκλιση και σε μικρότερη ικανότητα γενίκευσης. Συνεπώς κρίθηκε σκόπιμο να χρησιμοποιηθεί στην παρούσα εργασία και υλοποιήθηκε με χρήση του **StandardScaler** της **sklearn**.

**Κανονικοποίηση (Min-Max Scaling)**

Το min-max scaling είναι μια πολύ χρήσιμη τεχνική όταν τα δεδομένα βρίσκονται σε διαφορετικές κλίμακες. Με αυτή την τεχνική μεταφέρονται όλες οι τιμές σε μια συγκεκριμένη κλίμακα ([0,1]) έτσι ώστε να διατηρούνται μεν οι διαφορές (αναλογικά) μεταξύ των δειγμάτων ενός τύπου εισόδου αλλά μεταξύ δειγμάτων διαφορετικών τύπων εισόδων που βρίσκονται αρχικά σε διαφορετικά εύρη τιμών (π.χ. φύλο και βάρος) να υπάρξει εν τέλει μία ενιαία κλίμακα τιμών ώστε να μην δίνεται αρχικά περισσότερο βάρος σε κάποιες εισόδους από άλλες. Στη συγκεκριμένη εργασία το dataset που δίνεται περιλαμβάνει τιμές που βρίσκονται σε πολύ διαφορετικές κλίμακες και κατά συνέπεια κρίθηκε απαραίτητο να χρησιμοποιηθεί min-max scaling (υλοποιήθηκε με τον **MinMaxScaler** της **sklearn**). Ωστόσο η τεχνική αυτή είναι ευαίσθητη στην παρουσία ακραίων τιμών (outliers) καθώς τείνει να συμπιέσει τις συνηθισμένες τιμές (inliers) σε μικρό κομμάτι του εύρους τιμών. Αυτός είναι ακόμα ένας λόγος για τον οποίο πρέπει να εφαρμοστεί κεντράρισμα στα δεδομένα.

**Τυποποίηση (Standarization)**

Η τυποποίηση χρησιμοποιείται για να μετατρέψει τα δεδομένα ώστε αυτά να έχουν μέση τιμή μηδέν και διακύμανση ένα. Ωστόσο η τυποποίηση προϋποθέτει ότι τα δεδομένα έχουν Gaussian κατανομή ή κάτι κοντά σε αυτήν. Από την στιγμή που δεν γνωρίζουμε τίποτα για την κατανομή των δεδομένων μας δεν κρίνεται σκόπιμο να χρησιμοποιήσουμε αυτή την τεχνική ειδικά καθώς και αυτή είναι ευαίσθητη στην παρουσία outliers.

1. Για την δημιουργία των 5 folds που απαιτούνται χρησιμοποιήθηκε η **StratifiedKFold** της **sklearn.** Η συγκεκριμένη συνάρτηση σε αντίθεση με την **KFold** διατηρεί σε κάθε fold την αρχική αναλογία δειγμάτων ανά κλάση.

**Α2. Επιλογή αρχιτεκτονικής**

1. **Επιλογή συνάρτησης κόστους**

**Διεντροπία (Cross Entropy)**

Η διεντροπία μετράει την διαφορά μεταξύ της κατανομής πιθανότητας των παραγόμενων αποτελεσμάτων του νευρωνικού δικτύου και της κατανομής πιθανότητας των επιθυμητών αποτελεσμάτων. Ελαχιστοποιώντας την διεντροπία το μοντέλο αυξάνει την πιθανότητα ανάθεσης σωστής κλάσης στα δεδομένα εισόδου. Αυτό είναι ένα πολύ χρήσιμο χαρακτηριστικό για προβλήματα ταξινόμησης τα οποία απαιτούν ακριβώς αυτό, δηλαδή την ανάθεση ετικετών σε σετ δεδομένων εισόδου. Για αυτόν το λόγο η διεντροπία χρησιμοποιείται κατά βάση σε προβλήματα ταξινόμησης και για αυτόν τον λόγο χρησιμοποιήθηκε και στην παρούσα εργασία. Συγκεκριμένα χρησιμοποιείται **Sparse Categorical Crossentropy** επειδή χρησιμοποιούμε **integer encoding** για τις κλάσεις.

**Μέσο Τετραγωνικό Σφάλμα (MSE)**

Το Μέσο Τετραγωνικό Σφάλμα υπολογίζεται παίρνοντας το μέσο όρο των τετραγώνων των διαφορών των πραγματικών με των επιθυμητών τιμών. Δείχνει επί της ουσίας πόσο απέχουν οι προβλεπόμενες τιμές από τις πραγματικές. Χρησιμοποιείται κυρίως σε προβλήματα παλινδρόμησης.

**Ακρίβεια ταξινόμησης (Accuracy)**

Η ακρίβεια ταξινόμησης δείχνει απλώς το ποσοστό σωστών προβλέψεων ενός μοντέλου. Είναι χρήσιμη μετρική για να μετρήσουμε την απόδοση ενός μοντέλου σε προβλήματα ταξινόμησης αφού μας δείχνε ακριβώς πόσο καλά ταξινομήθηκαν τα δεδομένα εισόδου ωστόσο δεν είναι κατάλληλη για συνάρτηση κόστους καθώς δεν είναι διαφορίσιμη πράγμα απαραίτητο για τις συναρτήσεις κόστους. Επιπλέον η ακρίβεια ταξινόμησης δε δίνει κάποια πληροφορία σχετικά με τη “σιγουριά” με την οποία ταξινομούνται τα δεδομένα εισόδου. Π.χ. Ας πάρουμε την περίπτωση που ένα μοντέλο ταξινομεί τη φωτογραφία ενός σκύλου στην κλάση “σκύλος” με πιθανότητα 70% και στην κλάση “γάτα” με πιθανότητα 30% ενώ ένα άλλο μοντέλο ταξινομεί την ίδια φωτογραφία στην κλάση “σκύλος” με 90% και στην κλάση “γάτα” με 10%. Τα δύο μοντέλα θα έχουν την ίδια ακρίβεια ταξινόμησης αφού και τα δύο εν τέλει ταξινομούν σωστά την φωτογραφία όμως το δεύτερο είναι πολύ πιο “σίγουρο” για την ταξινόμηση που έκανε κάτι το οποίο φαίνεται μόνο στην διεντροπία του και όχι στην ακρίβεια ταξινόμησης.

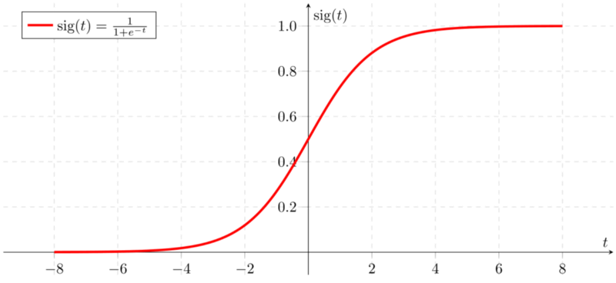
1. **Επιλογή αριθμού νευρώνων στο επίπεδο εξόδου**

Λόγω του ότι έχουμε πρόβλημα **Multiclass Classifaction** δηλαδή θέλουμε να ταξινομήσουμε τα δεδομένα εισόδου σε μια σειρά από αμοιβαία αποκλειόμενες ετικέτες θα χρησιμοποιήσουμε ένα νευρώνα ανά κλάση (άρα το πλήθος των νευρώνων ισούται με το πλήθος των κλάσεων δηλαδή 5) . Ο κάθε νευρώνας επιστρέφει την πιθανότητα με την οποία ταξινομείται η είσοδος στην κλάση στην οποία αντιστοιχεί.

1. **Επιλογή συνάρτησης ενεργοποίησης για τους κρυφούς κόμβους**

Η συνάρτηση ενεργοποίησης των κρυφών νευρώνων πρέπει να είναι μη γραμμική καθώς μια γραμμική συνάρτηση ενεργοποίησης θα σήμαινε απλώς ότι οι έξοδοι των νευρώνων είναι ένας γραμμικός συνδυασμός των εισόδων τους καθιστώντας αχρείαστη την ύπαρξη κρυφών επιπέδων νευρώνων. Βασική προϋπόθεση της συνάρτησης ενεργοποίησης είναι να είναι διαφορίσιμη καθώς η παράγωγός της χρησιμοποιείται από τον Αλγόριθμο Πίσω Διάδοσης για τον υπολογισμό των βαρών. Συνήθως για τους κρυφούς κόμβους χρησιμοποιούνται τρεις συναρτήσεις:

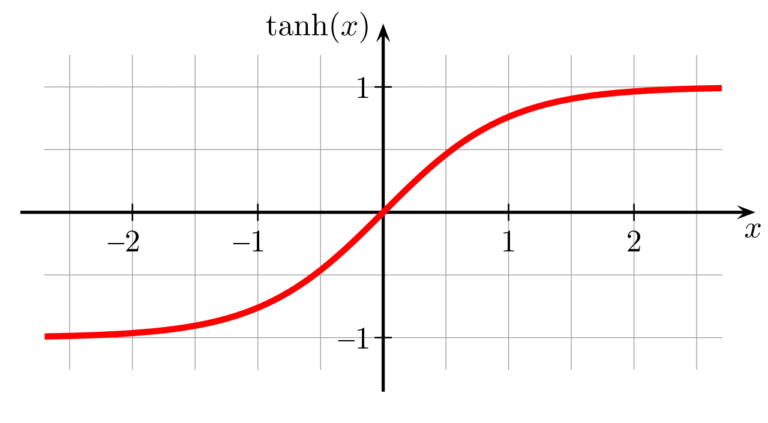
**Λογιστική συνάρτηση**



Πηγή: https://www.aitude.com/

Η λογιστική συνάρτηση χρησιμοποιείται ως συνάρτηση ενεργοποίησης για τους κρυφούς νευρώνες καθώς πληροί τα κριτήρια που περιγράψαμε παραπάνω ωστόσο παρουσιάζει δύο βασικά προβλήματα. Πρώτον ότι δεν έχει κέντρο το μηδέν πράγμα το οποίο μπορεί να οδηγήσει σε ταλαντώσεις στη μεταβολή των βαρών. Δεύτερον ότι όταν η έξοδος του νευρώνα βρίσκεται στα όρια της συνάρτησης η παράγωγος τείνει στο μηδέν με αποτέλεσμα ο νευρώνας να μη συνεισφέρει στην πίσω διάδοση του λάθους. Το φαινόμενο αυτό ονομάζεται gradient kill.

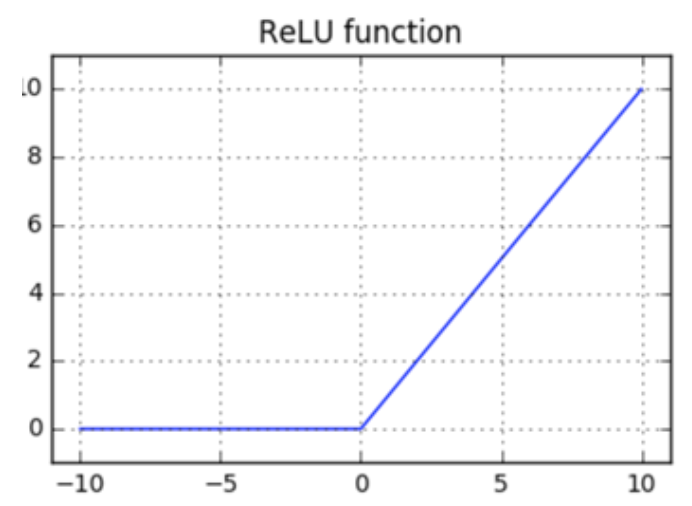
**Συνάρτηση Υπερβολικής Εφαπτομένης (Tanh)**



Πηγή: https://www.aitude.com

Πρόκειται για μαθηματικό μετασχηματισμό της λογιστικής. Προτιμάται έναντι της τελευταίας λόγω του ότι έχει κέντρο το μηδέν ωστόσο εμφανίζεται και και πάλι το φαινόμενο gradient kill.

**ReLU (Rectified Linear Unit)**



Πηγή: https://www.aitude.com

Η συγκεκριμένη συνάρτηση δεν παρουσιάζει το φαινόμενο gradient kill καθώς έχει εύρος τιμών [0, inf). Λόγω του ότι για αρνητικές εισόδους επιστρέφει μηδενική έξοδο, ορισμένοι νευρώνες δεν ενεργοποιούνται πάντα με αποτέλεσμα καλύτερη απόδοση και ευκολότερους υπολογισμούς. Επιπλέον επειδή πρόκειται για απλούστερη συνάρτηση μαθηματικά είναι και υπολογιστικά φθηνότερη. Για αυτούς τους λόγους χρησιμοποιείται γενικώς περισσότερο από τις προαναφερθείσες και επιλέχθηκε και για τη συγκεκριμένη εργασία.

1. Για το επίπεδο εξόδου επιλέχθηκε η συνάρτηση **Softmax**. Ο λόγος για αυτό είναι όπως προαναφέρθηκε ότι έχουμε πρόβλημα **Multiclass Clasification** και κάθε κόμβος εξόδου επιστρέφει την πιθανότητα η είσοδος να ανήκει σε μια κλάση. Συνεπώς θέλουμε μια συνάρτηση ενεργοποίησης η οποία να μας δίνει αυτές τις πιθανότητες οι οποίες να αθροίζουν στο 1. Αυτήν ακριβώς τη λειτουργία επιτελεί η **Softmax** και για αυτό το λόγο χρησιμοποιείται σε τέτοιου είδους προβλήματα.
2. **Πειράματα με τον αριθμό των κρυφών νευρώνων**

Για την εκπαίδευση του νευρωνικού δικτύου χρησιμοποιήθηκε batch μεγέθους 128 και ο μέγιστος αριθμός κύκλων εκπαίδευσης ανά fold ορίστηκε στις 600. Ως μέθοδος βελτιστοποίησης χρησιμοποιήθηκε το **gradient descent** με ρυθμό μάθησης η=0.001 και σταθερά ορμής m=0 (Γενικευμένος Κανόνας Δέλτα). Εφαρμόστηκε επιπλέον η μέθοδος πρόωρου τερματισμού με κατάλληλο κριτήριο.

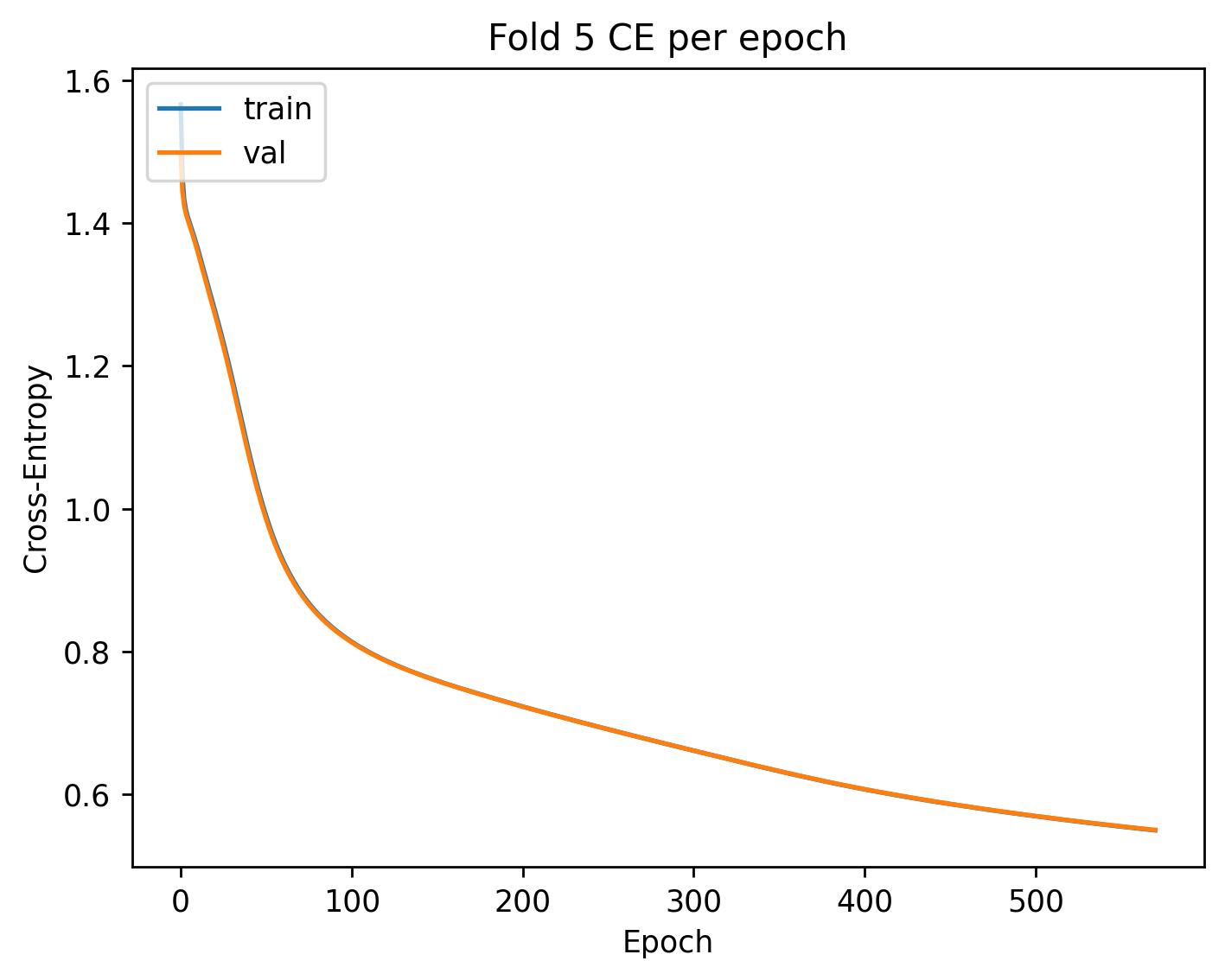
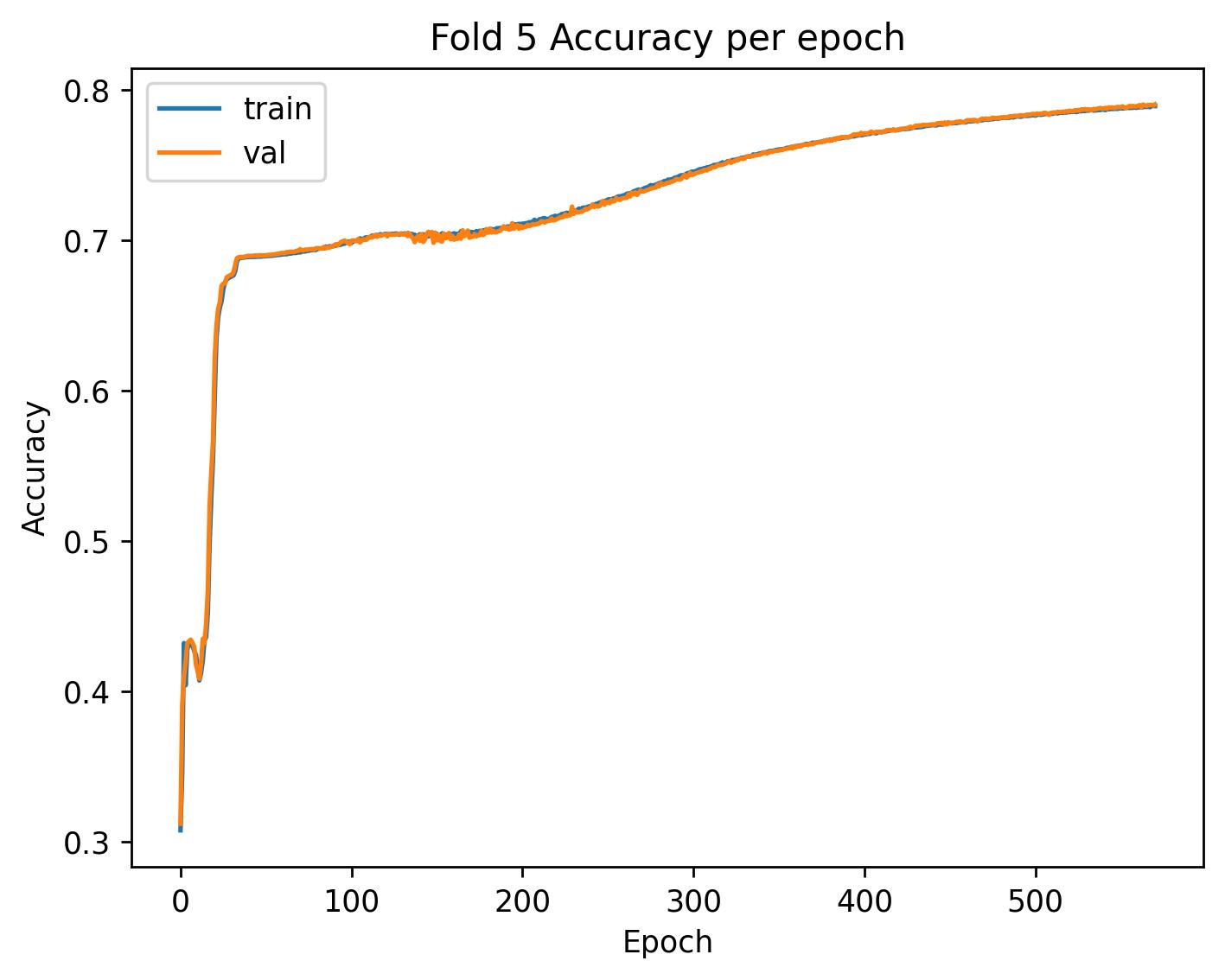
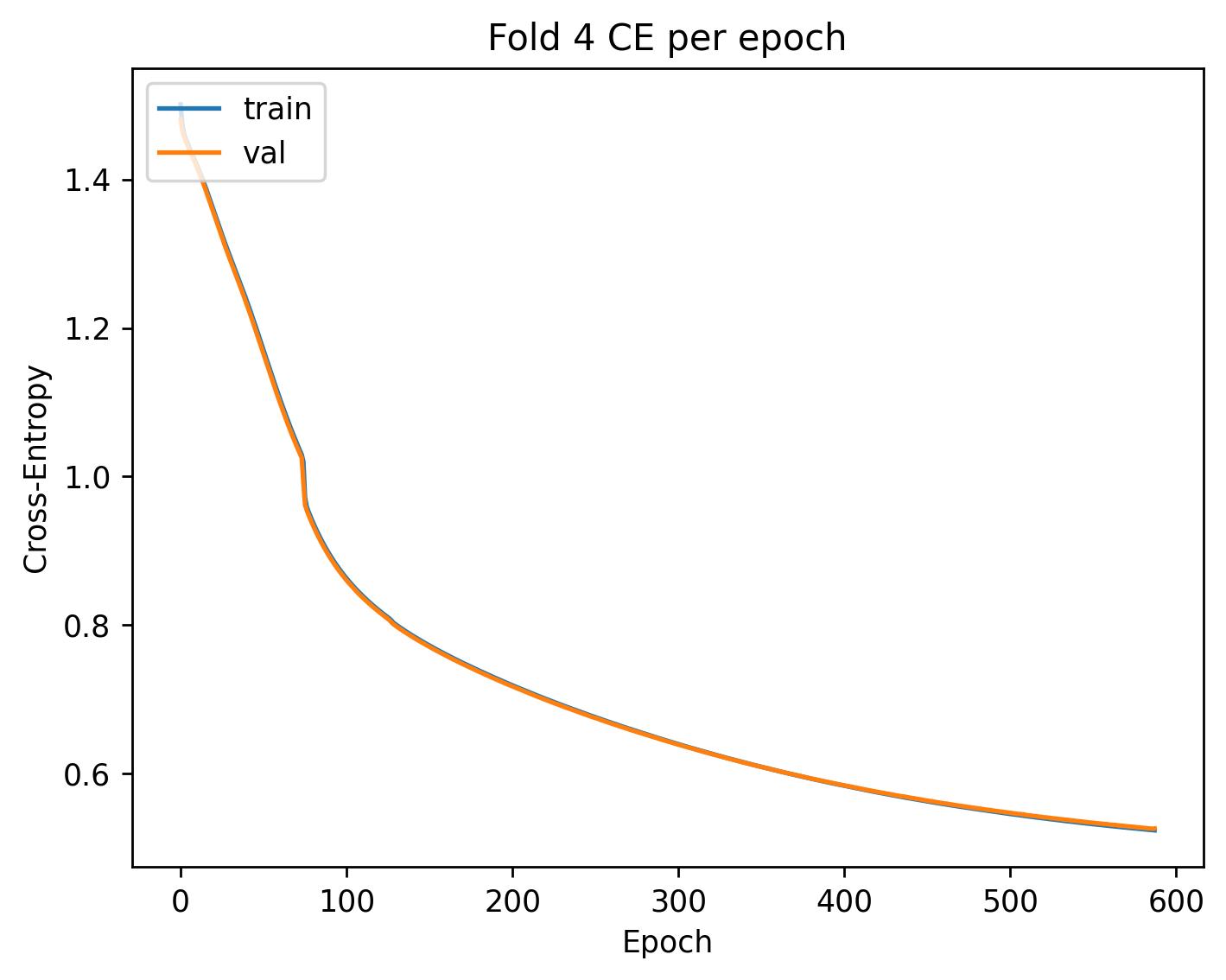
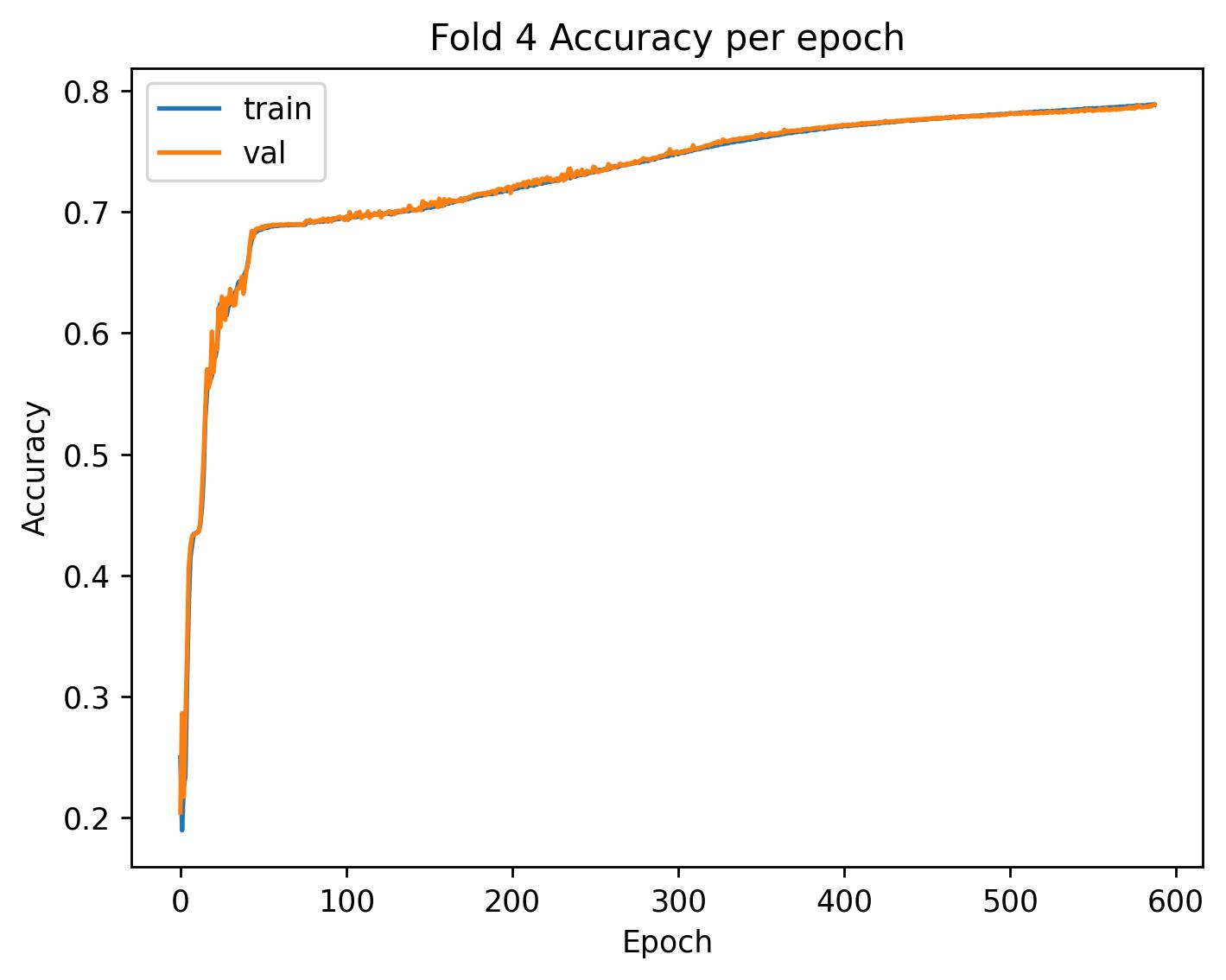
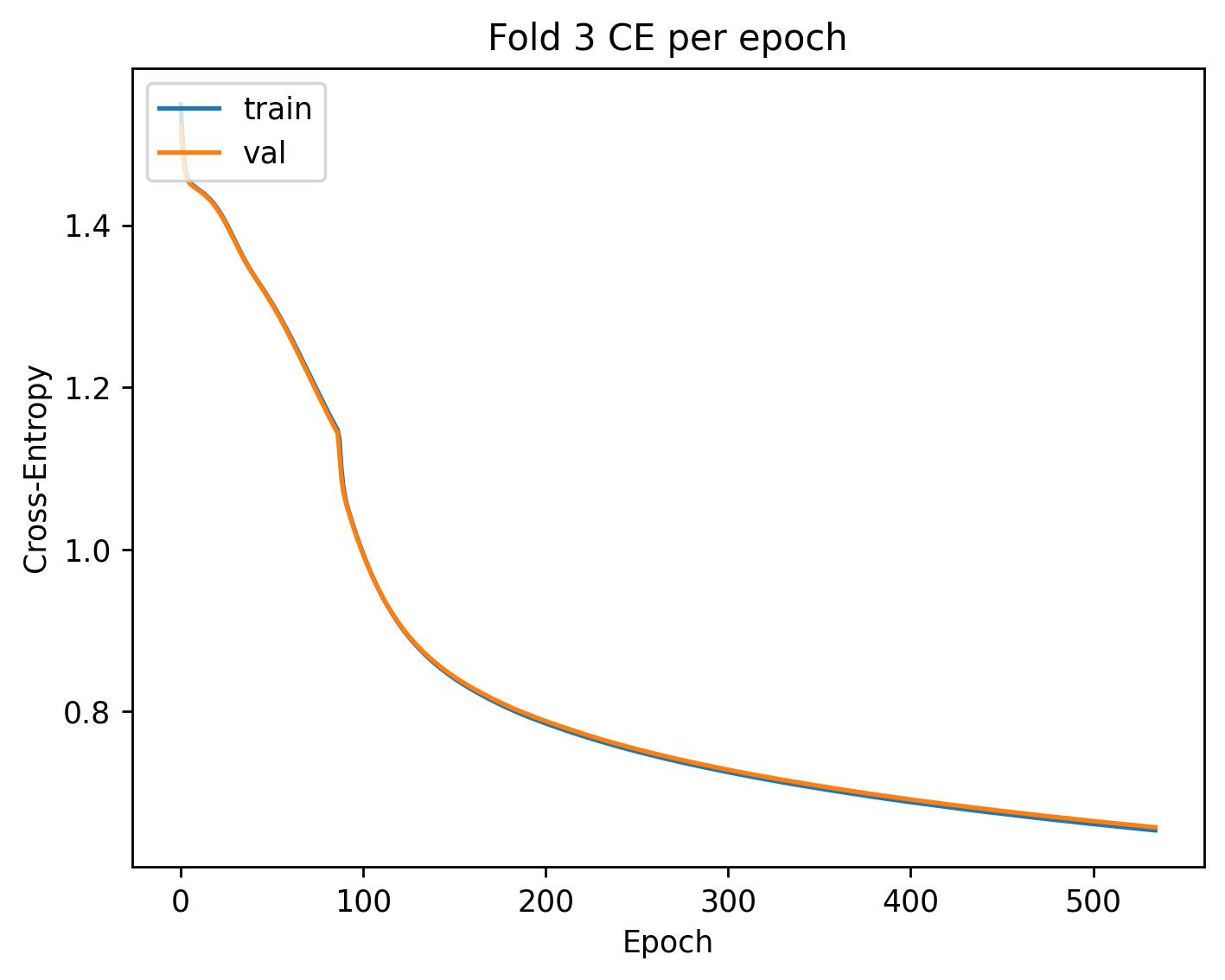
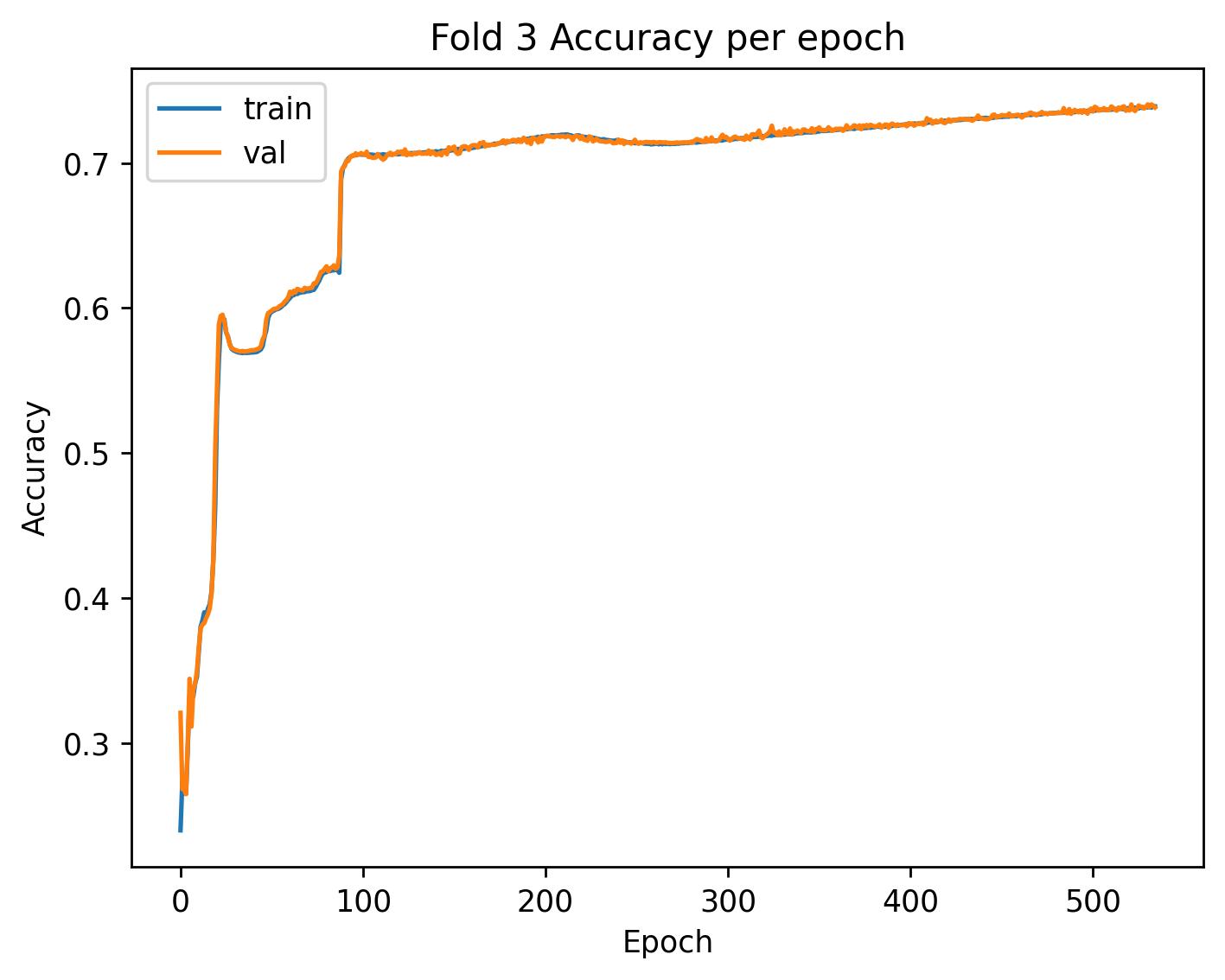
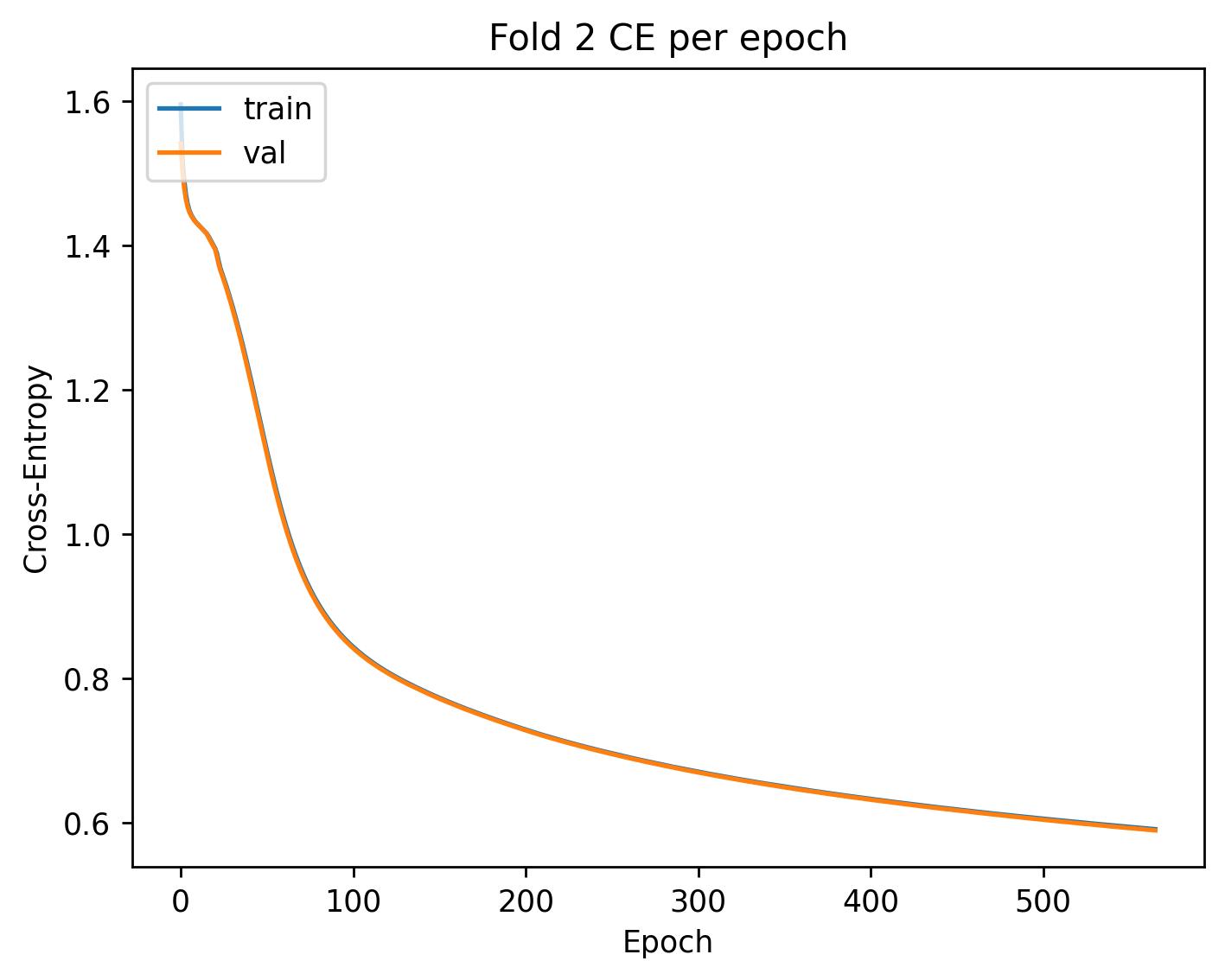
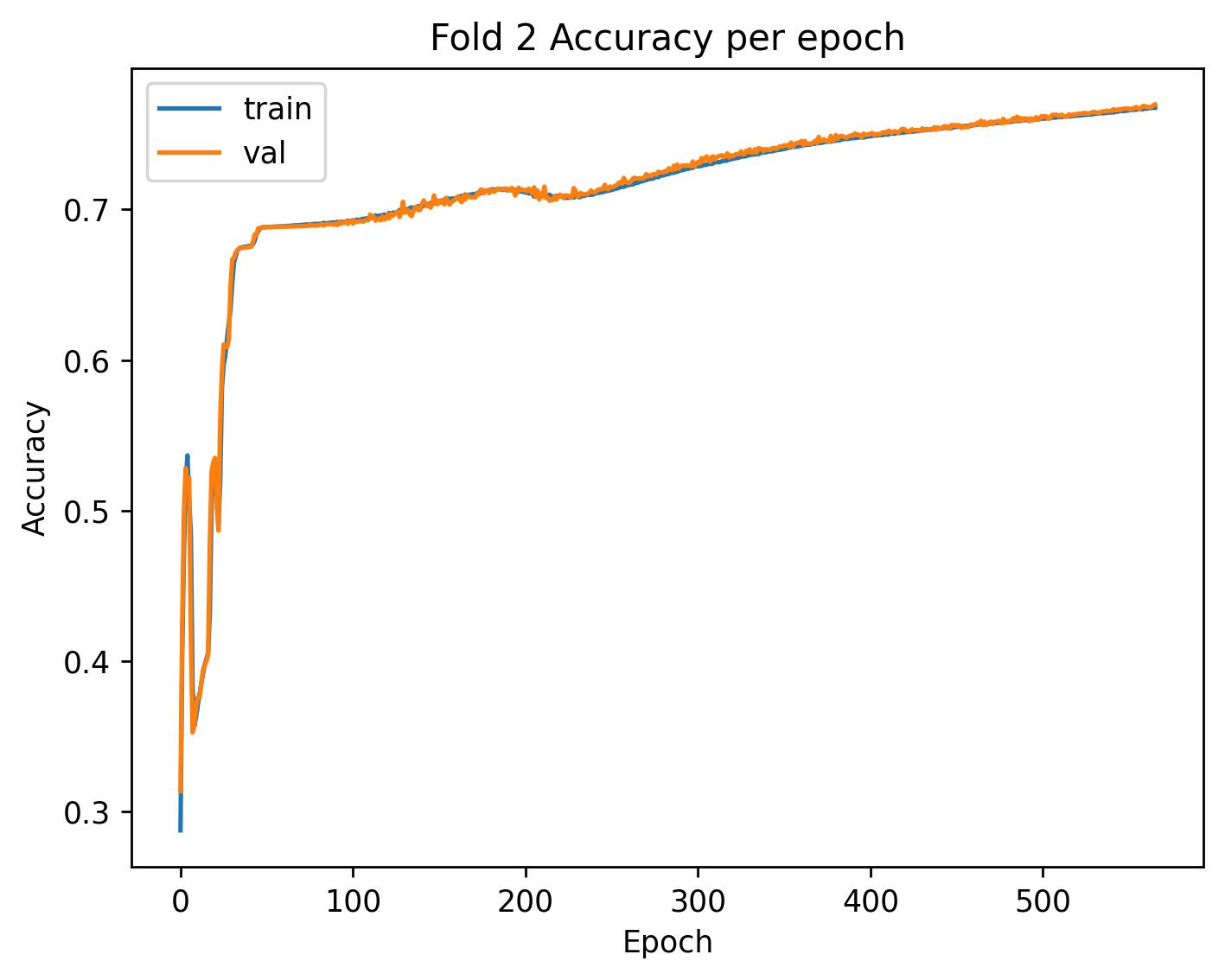
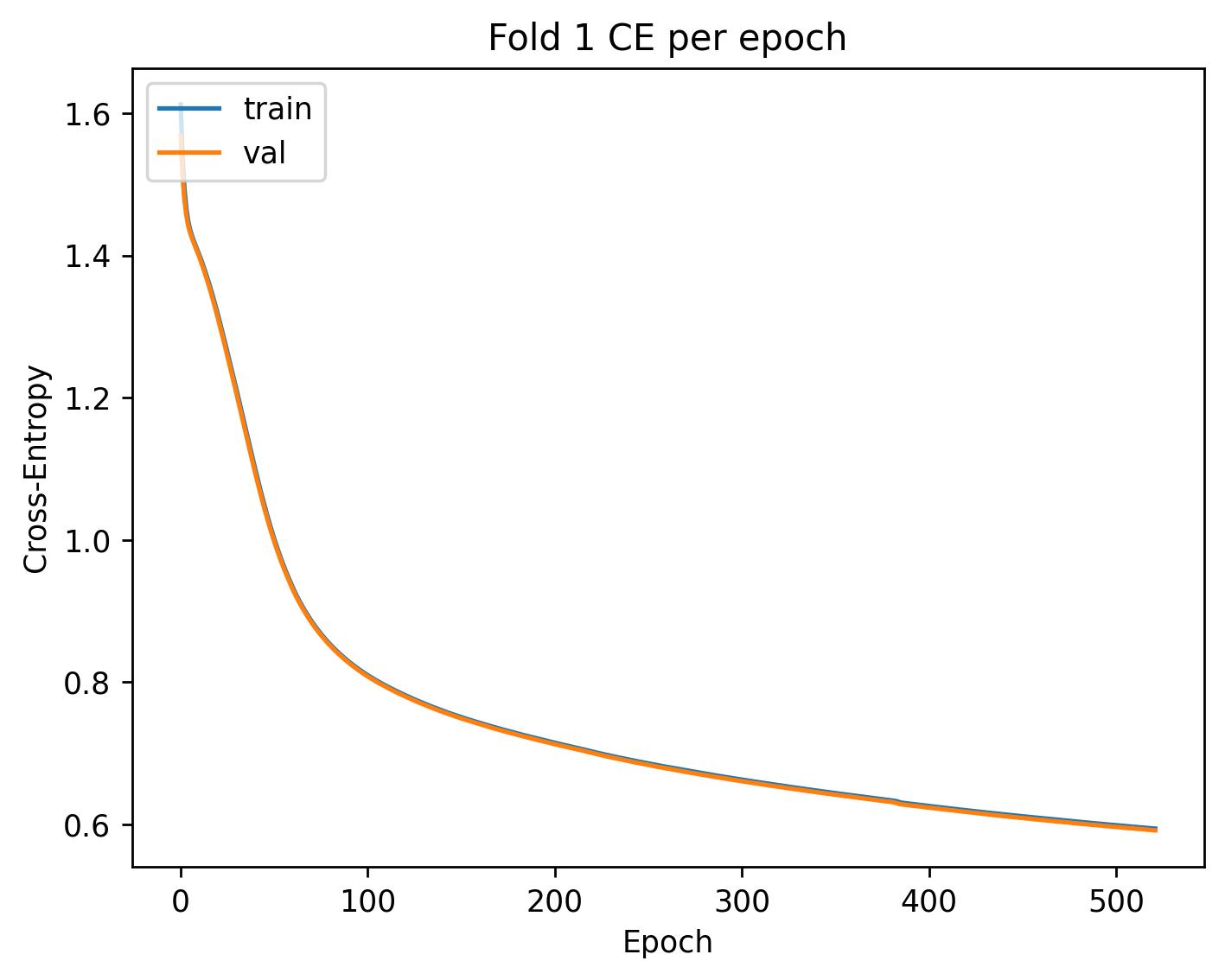
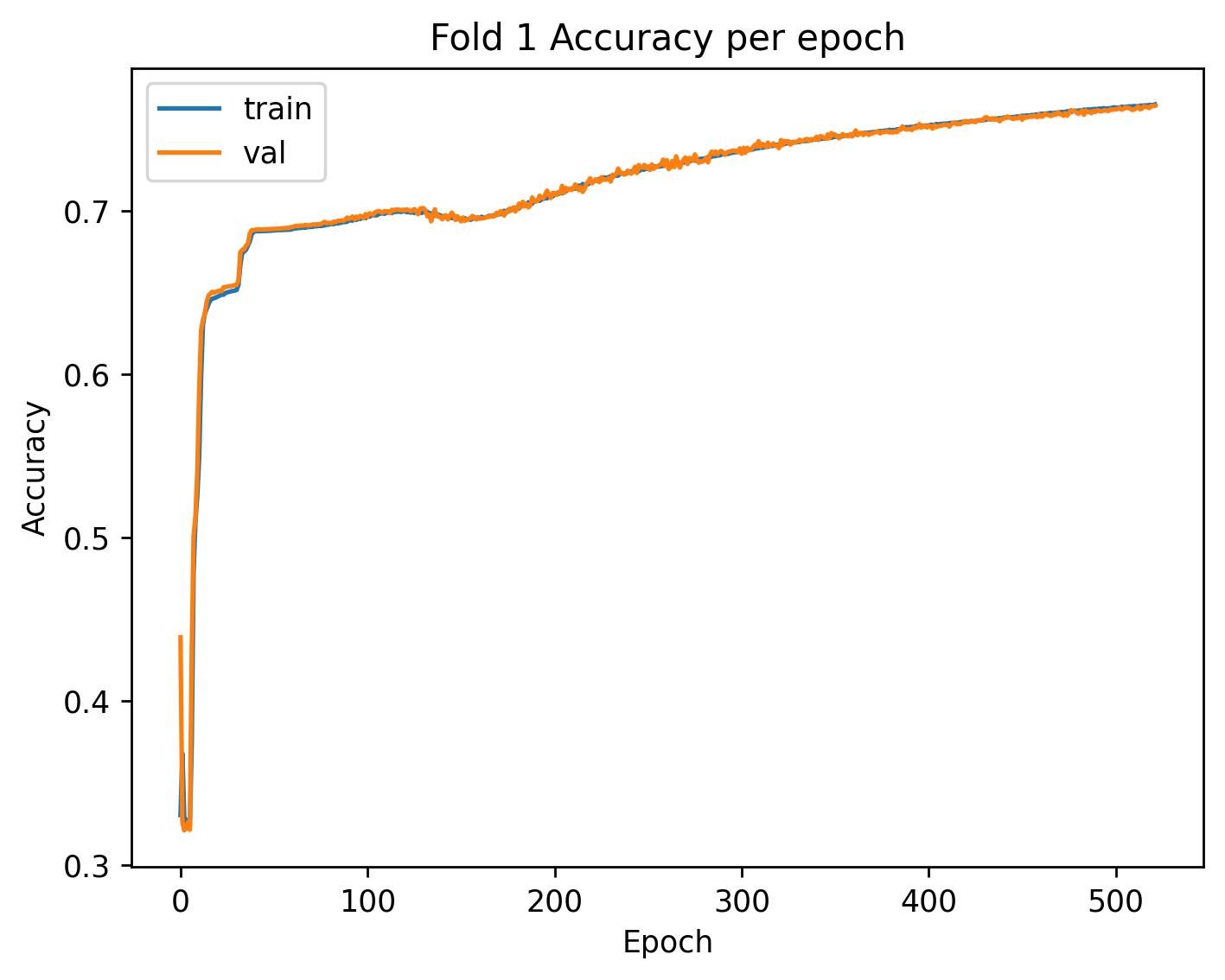
**Αποτελέσματα**

**Πίνακας μέσω τιμών μετρικών**

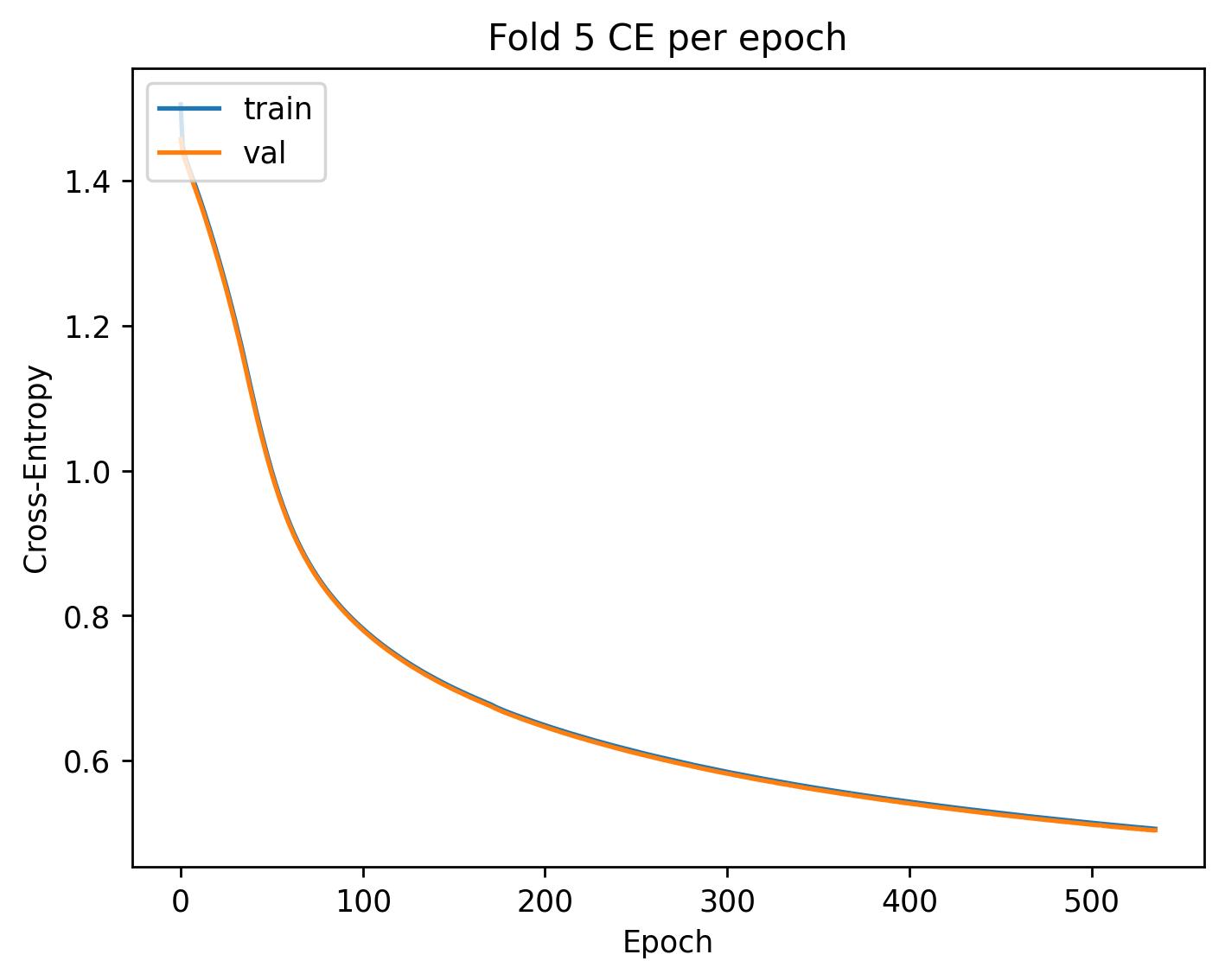
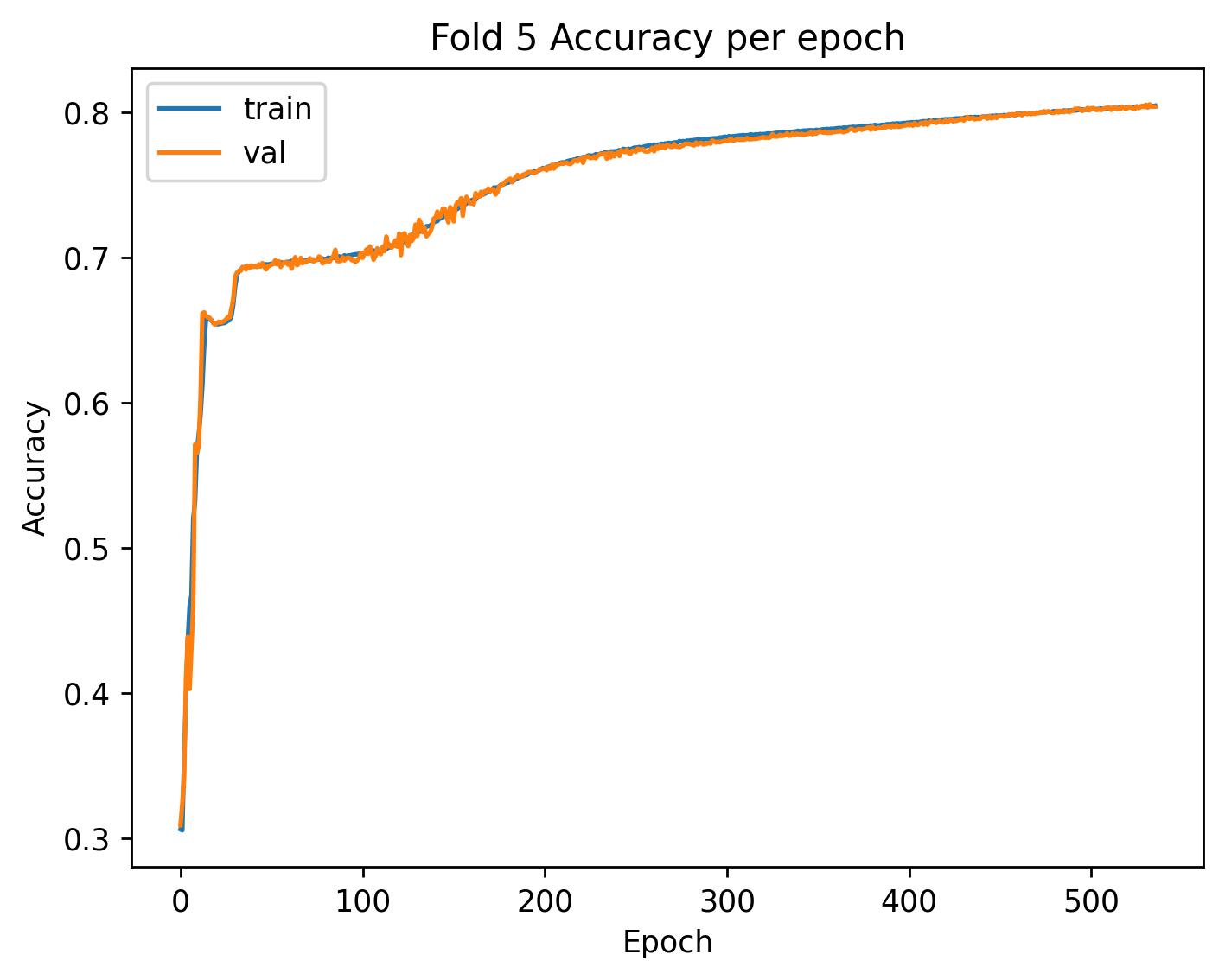
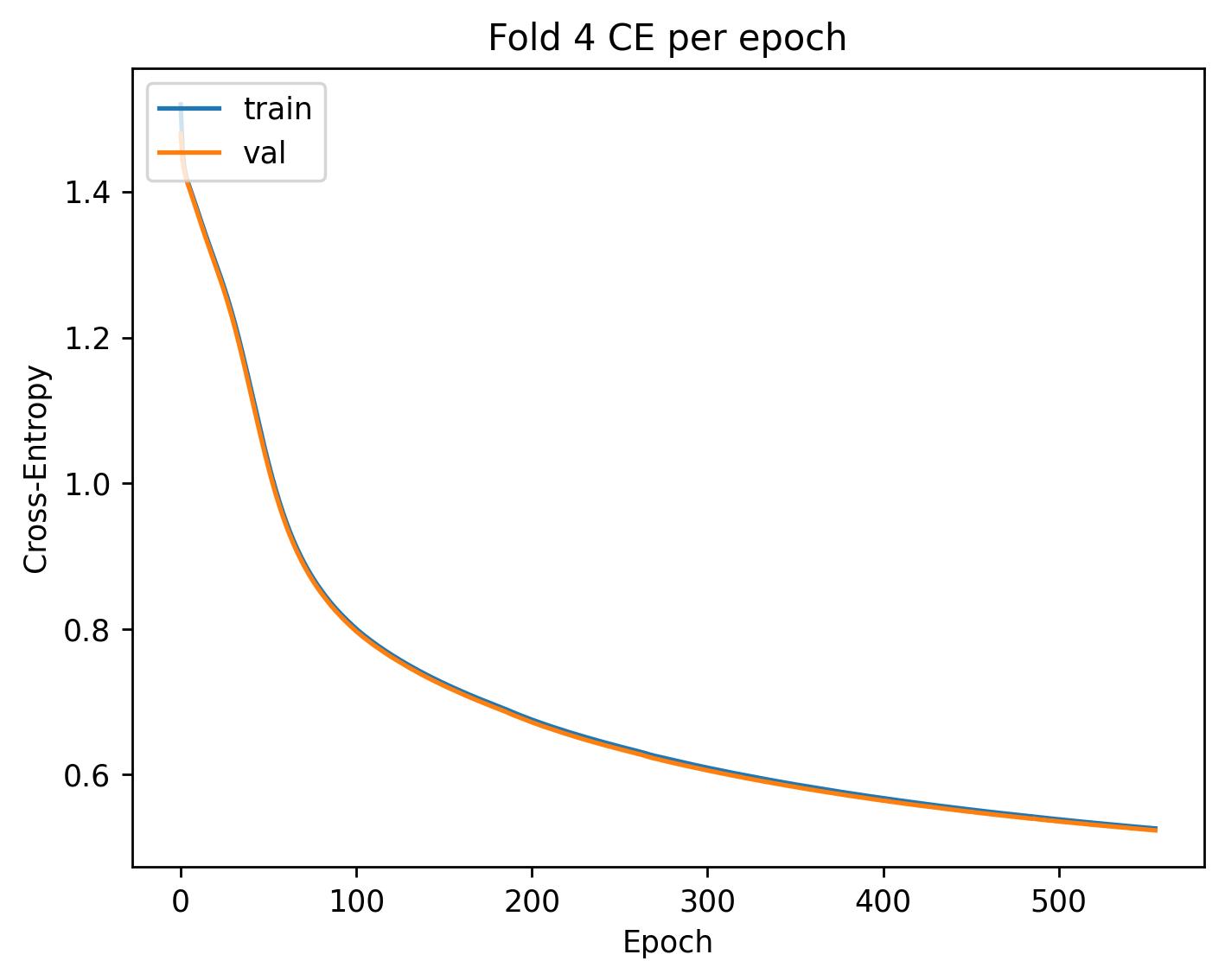
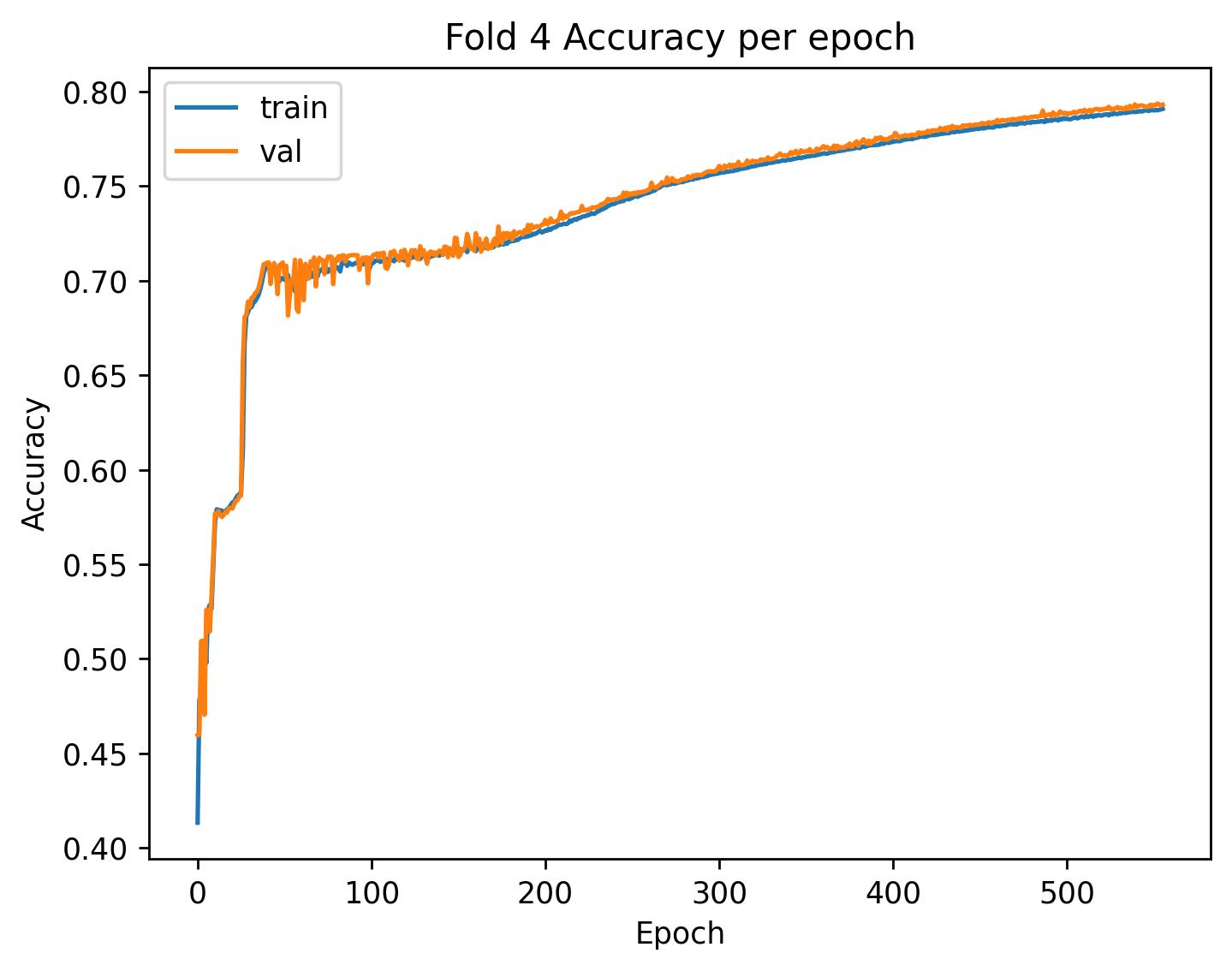
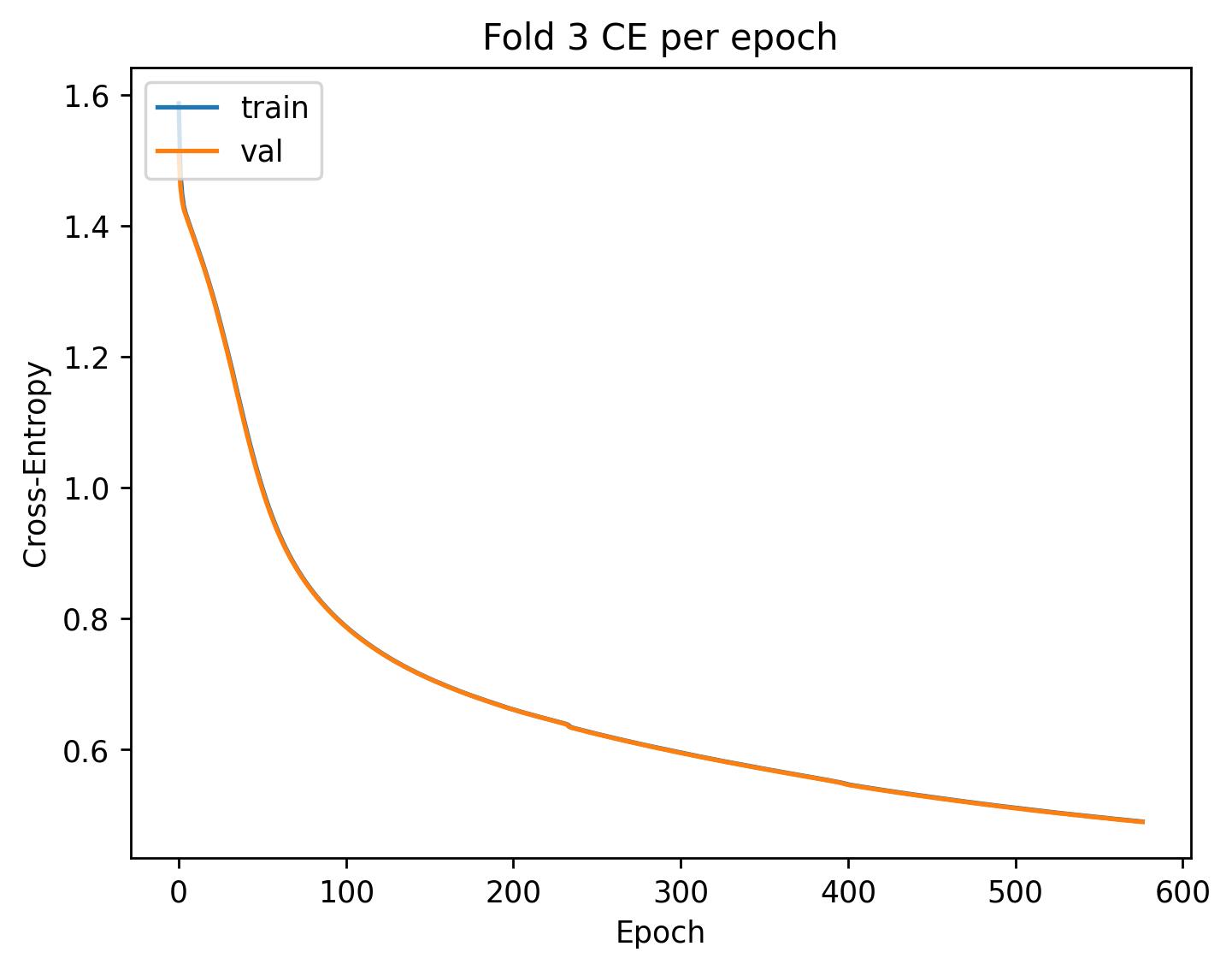
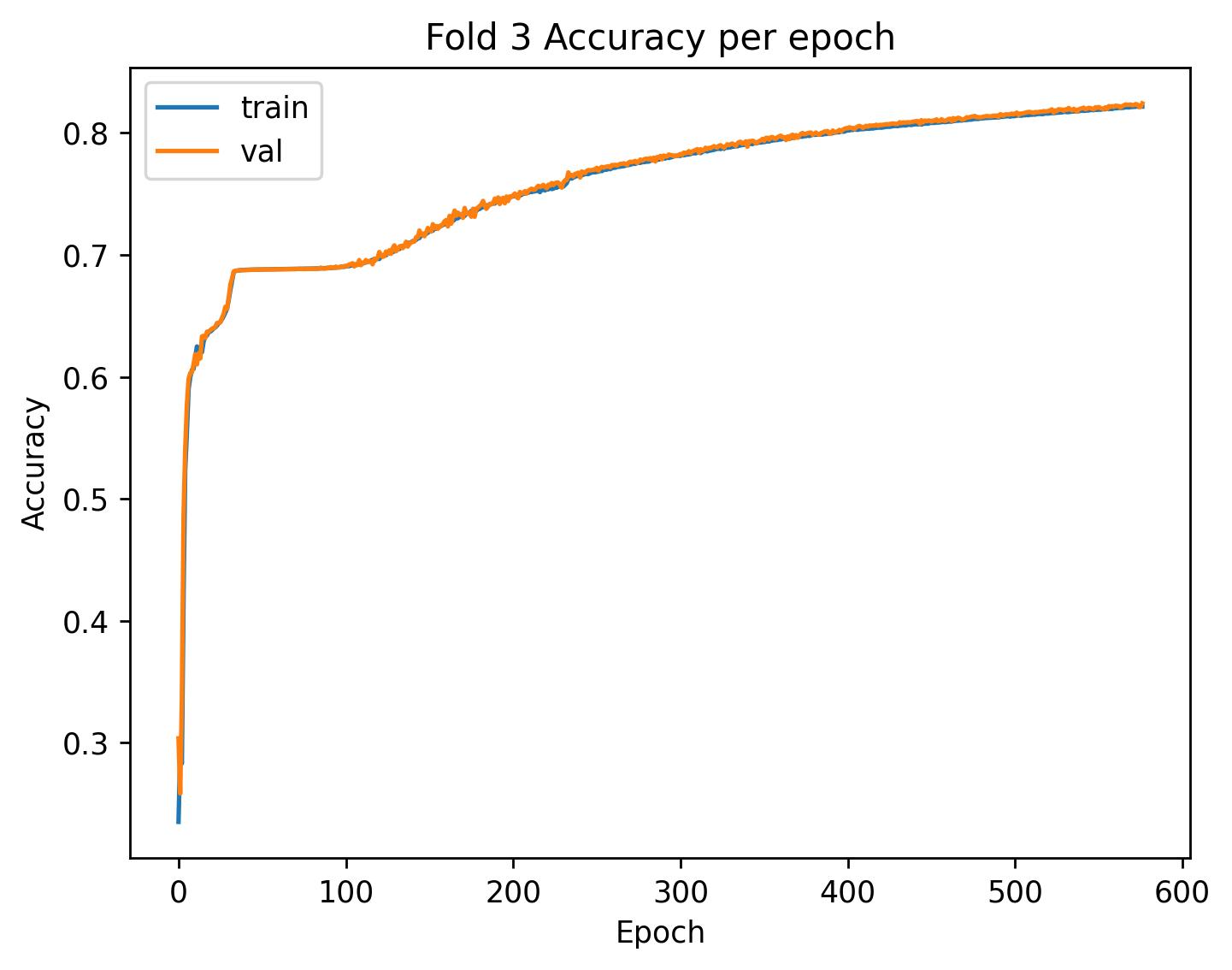
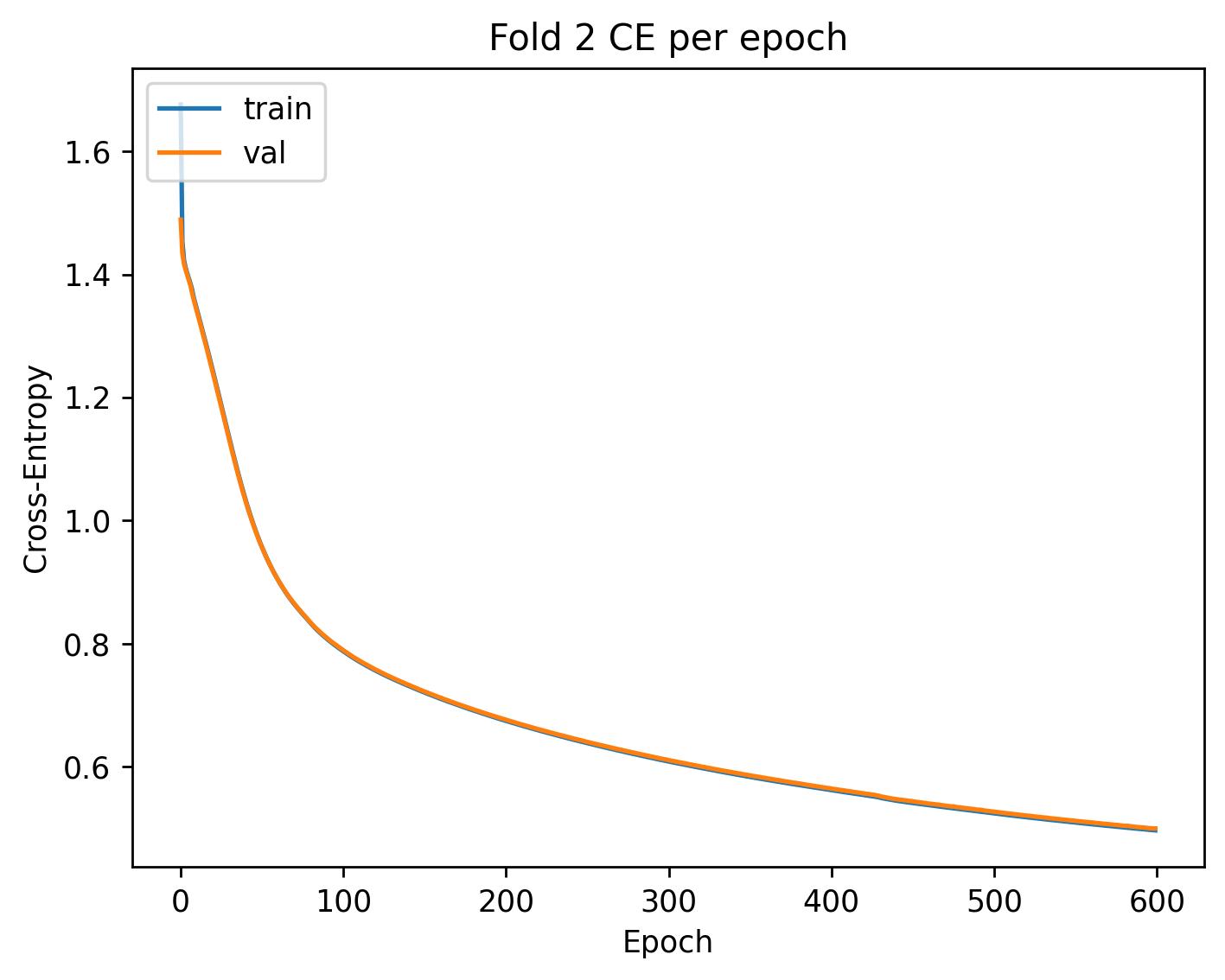
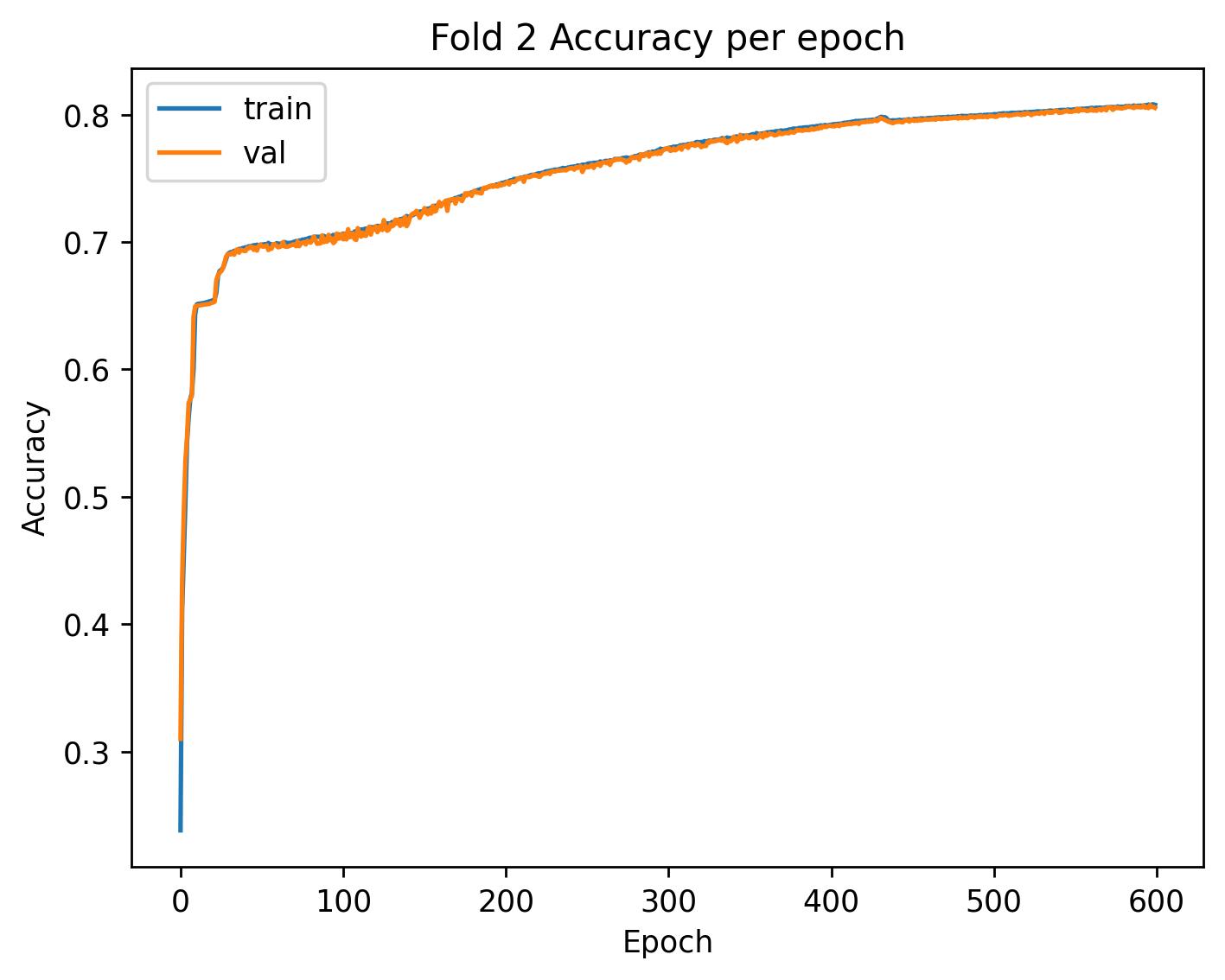
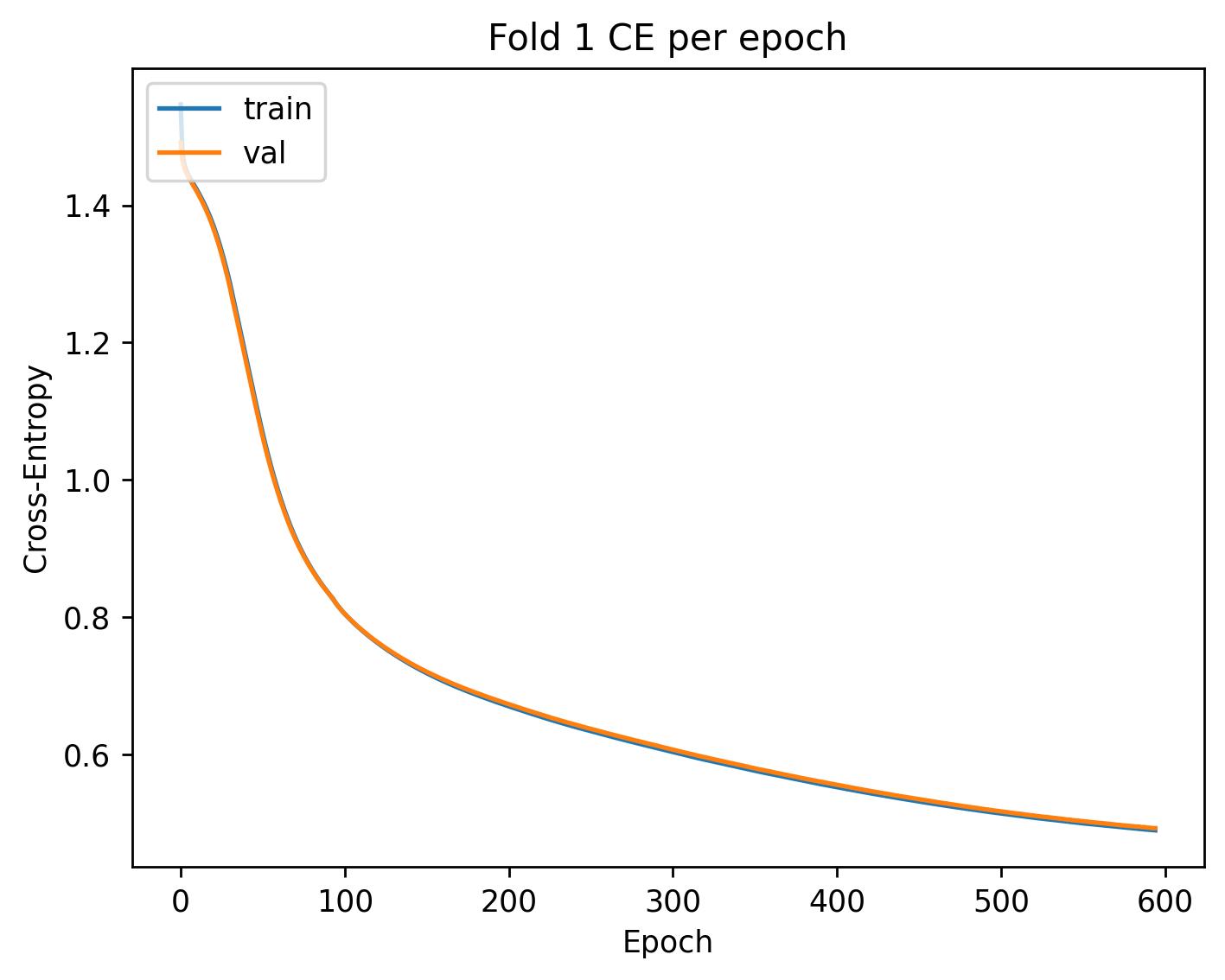
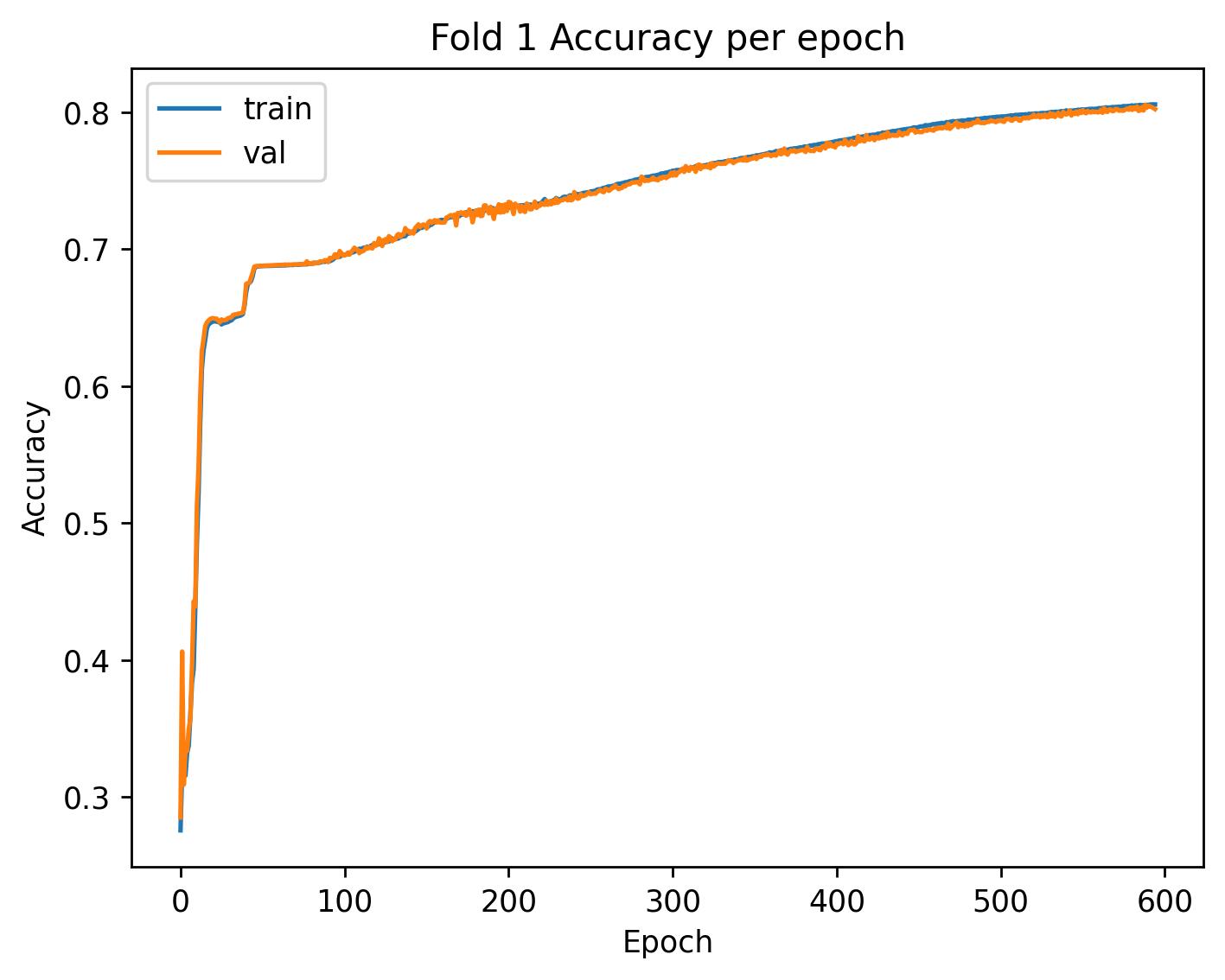
|  |  |  |  |
| --- | --- | --- | --- |
| **Αριθμός νευρώνων στο κρυφό επίπεδο** | **CE Loss** | **MSE** | **Accuracy** |
| H1 = 5 | 0.584 | 5.442 | 0.769 |
| H1 = 12 | 0.503 | 5.450 | 0.806 |
| H1 = 23 | 0.482 | 5.451 | 0.812 |

**Γραφικές παραστάσεις**

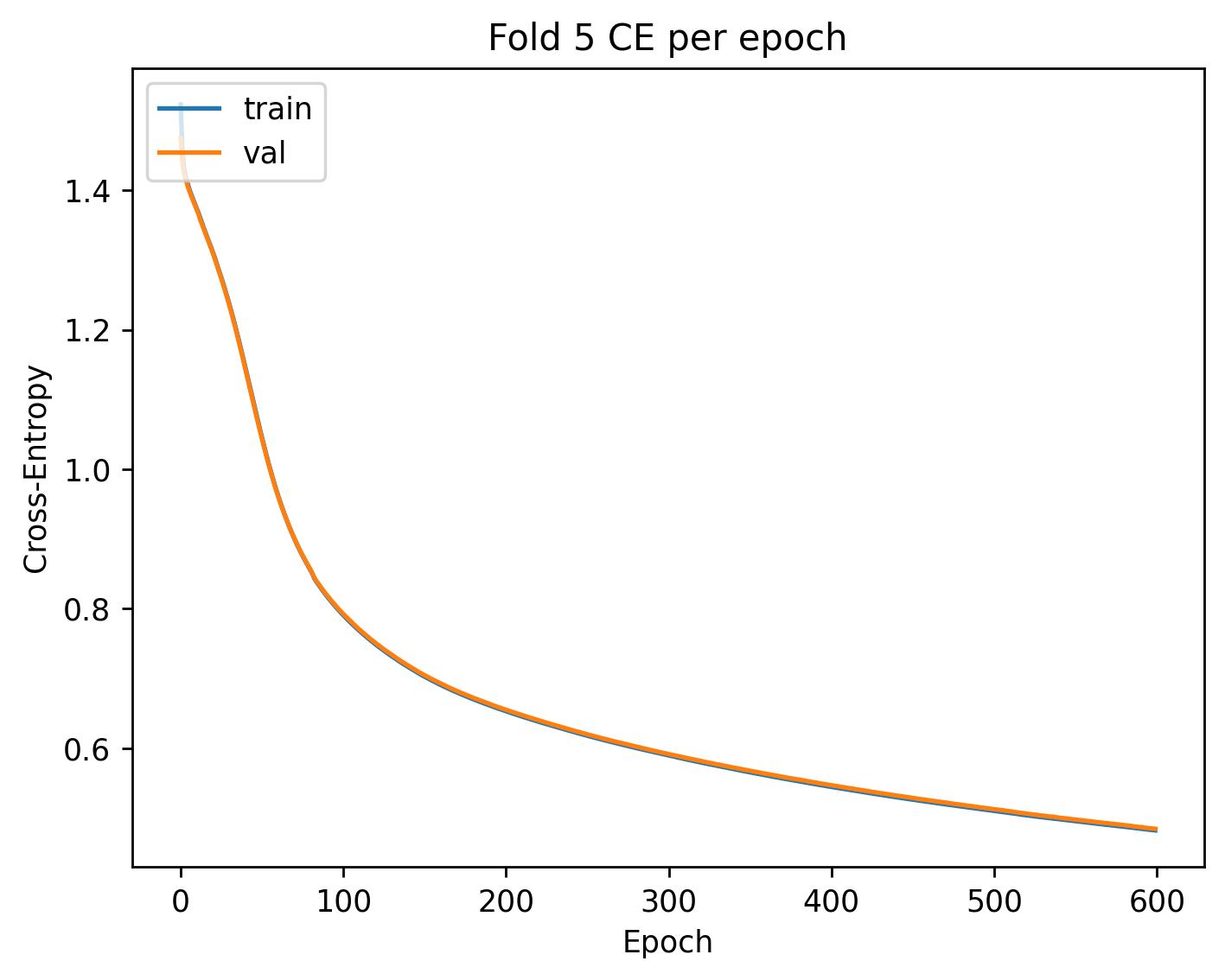
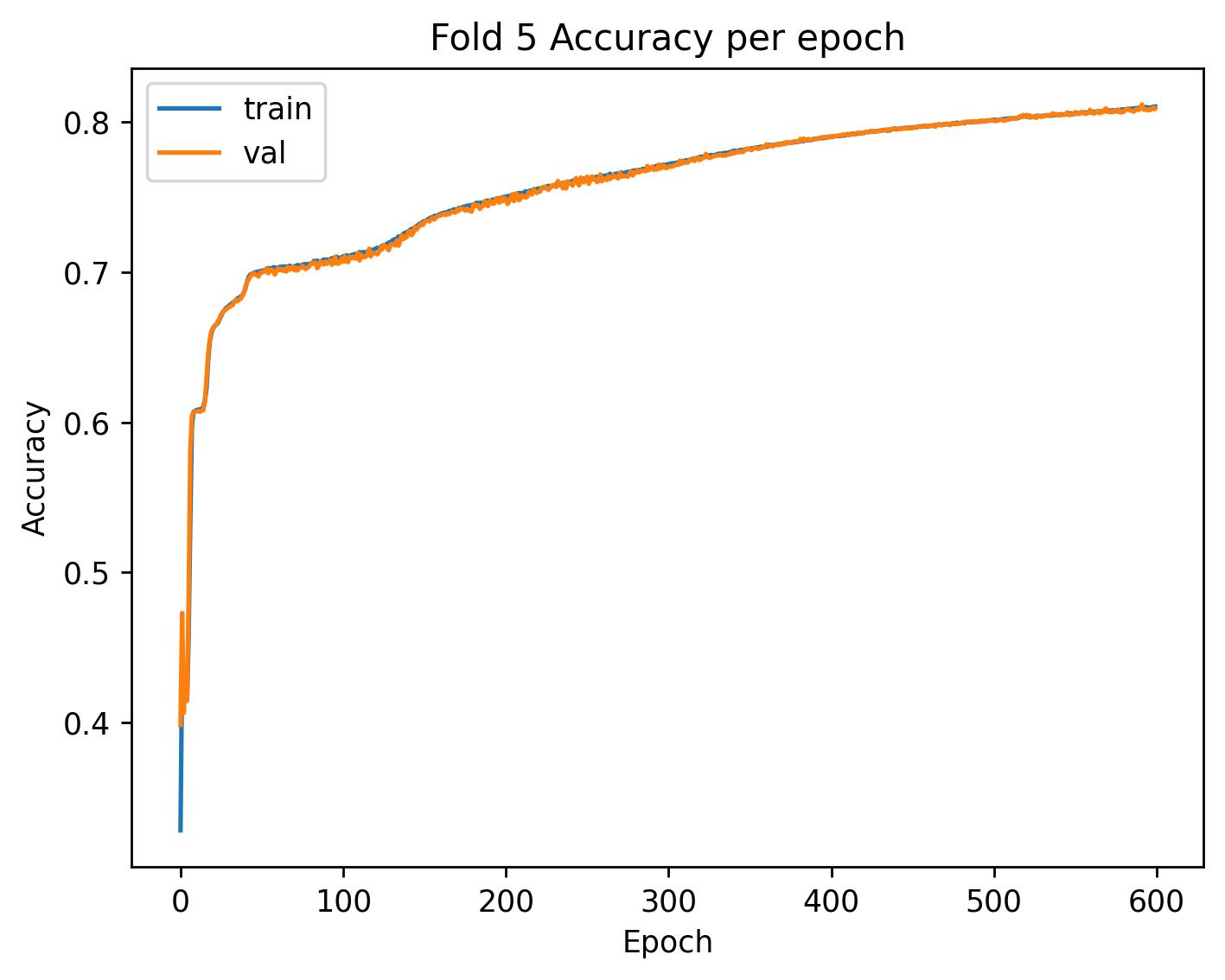
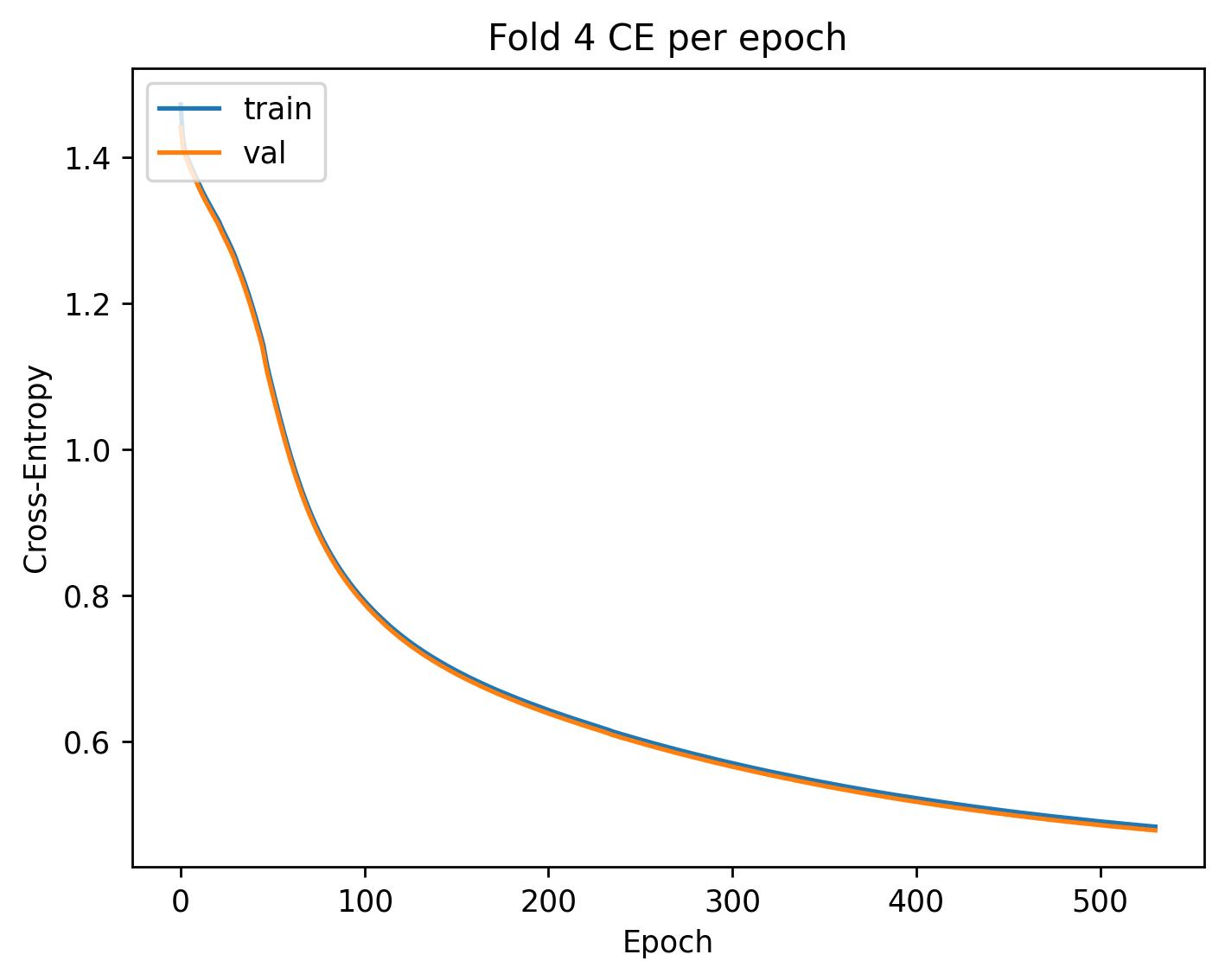
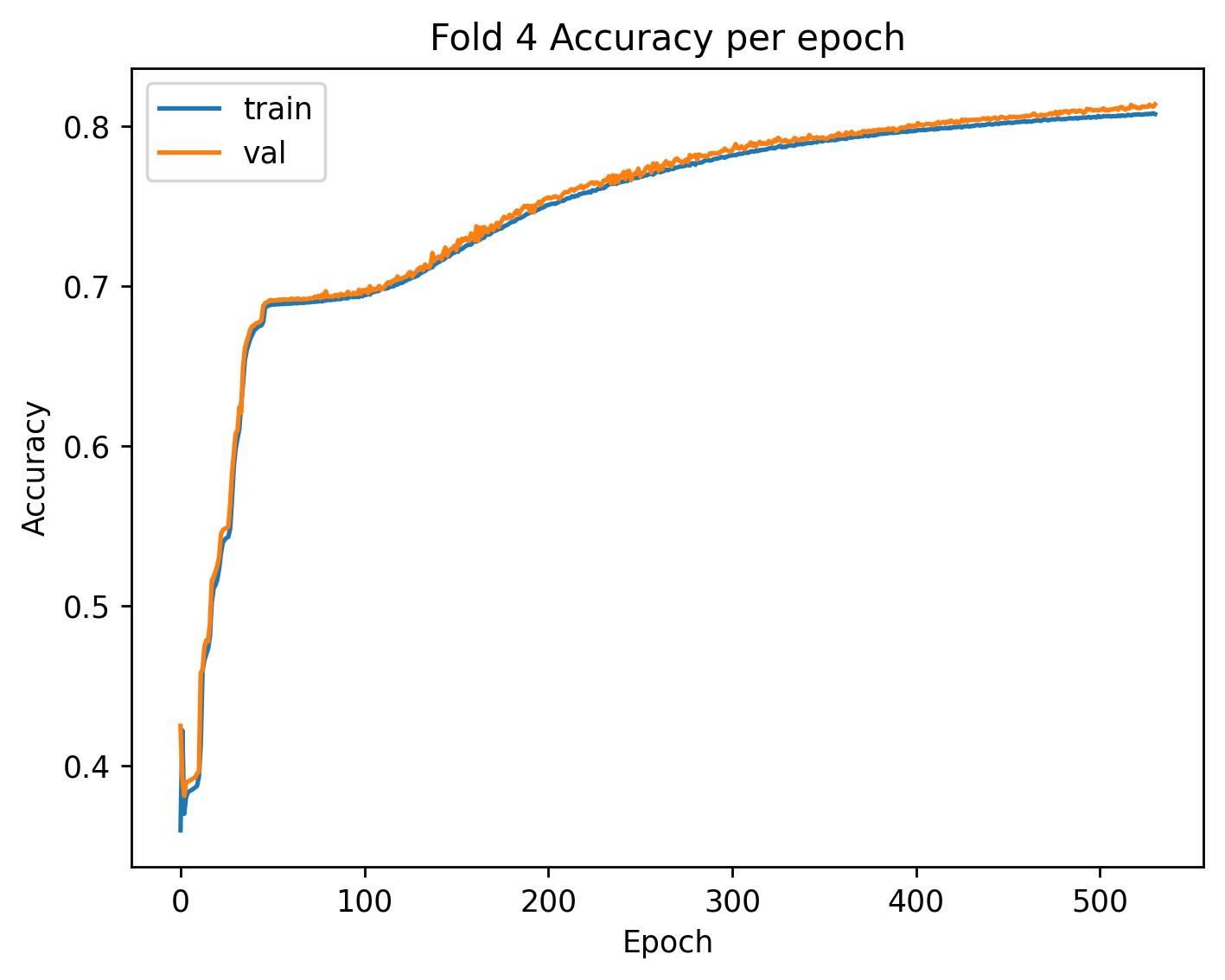
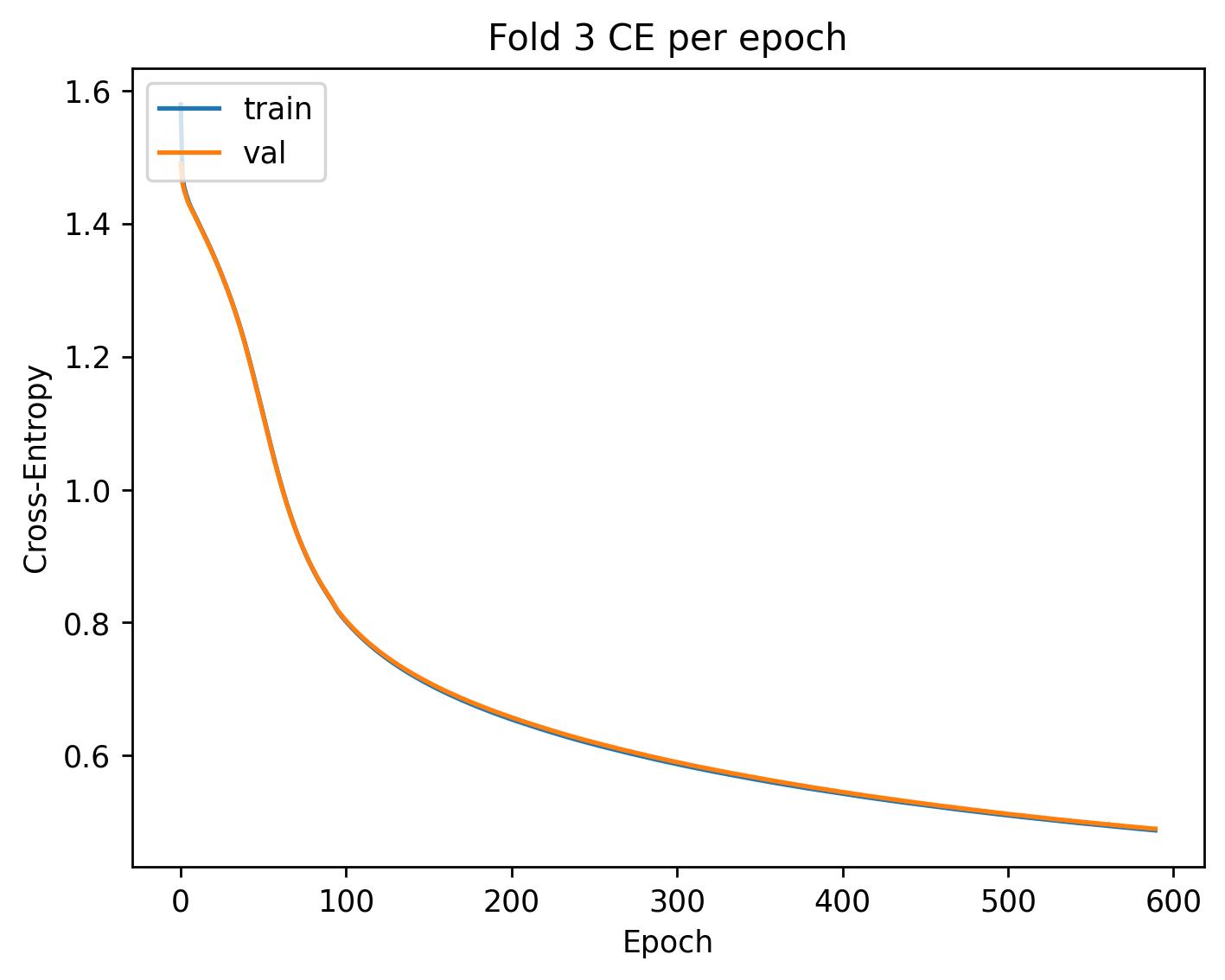
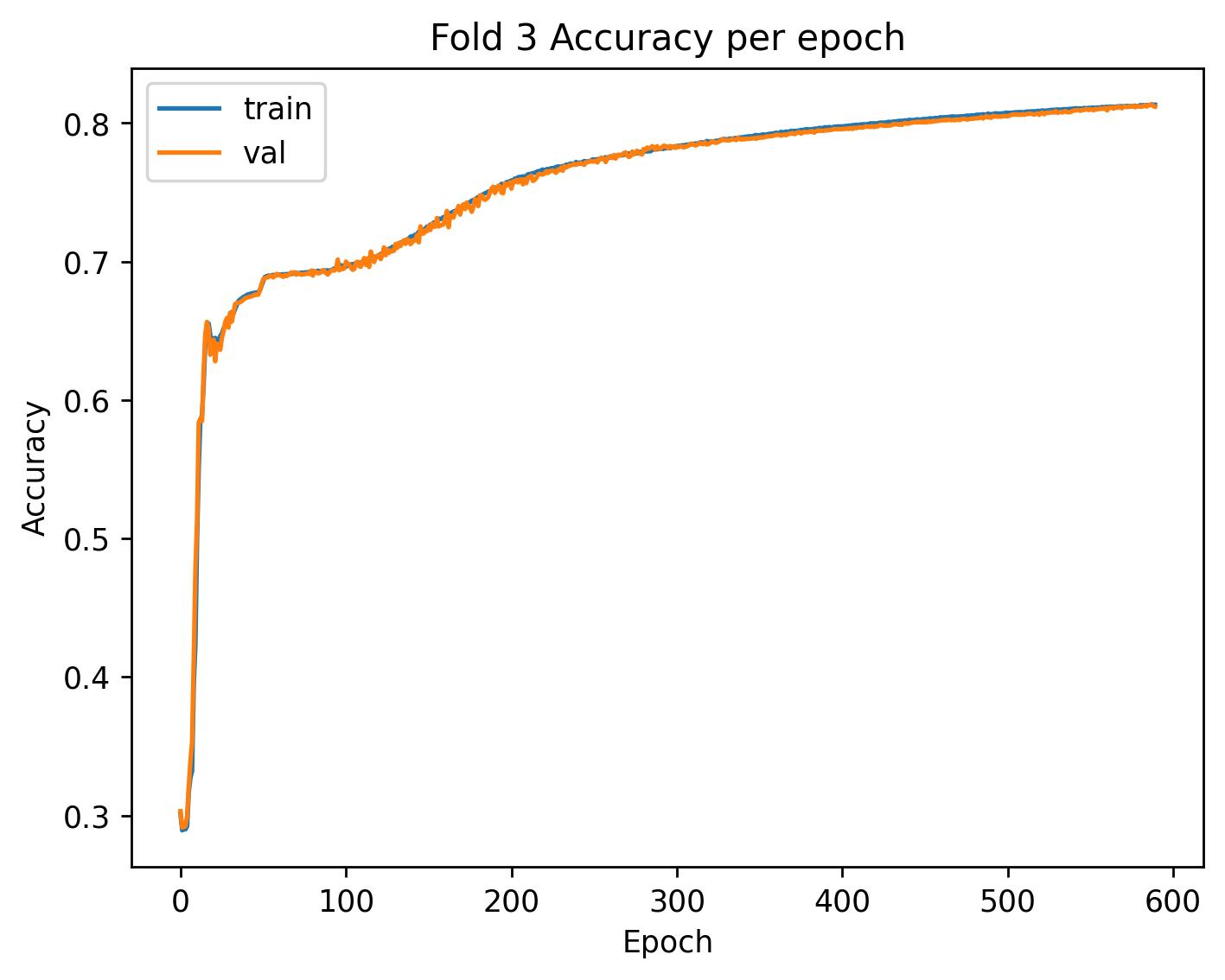
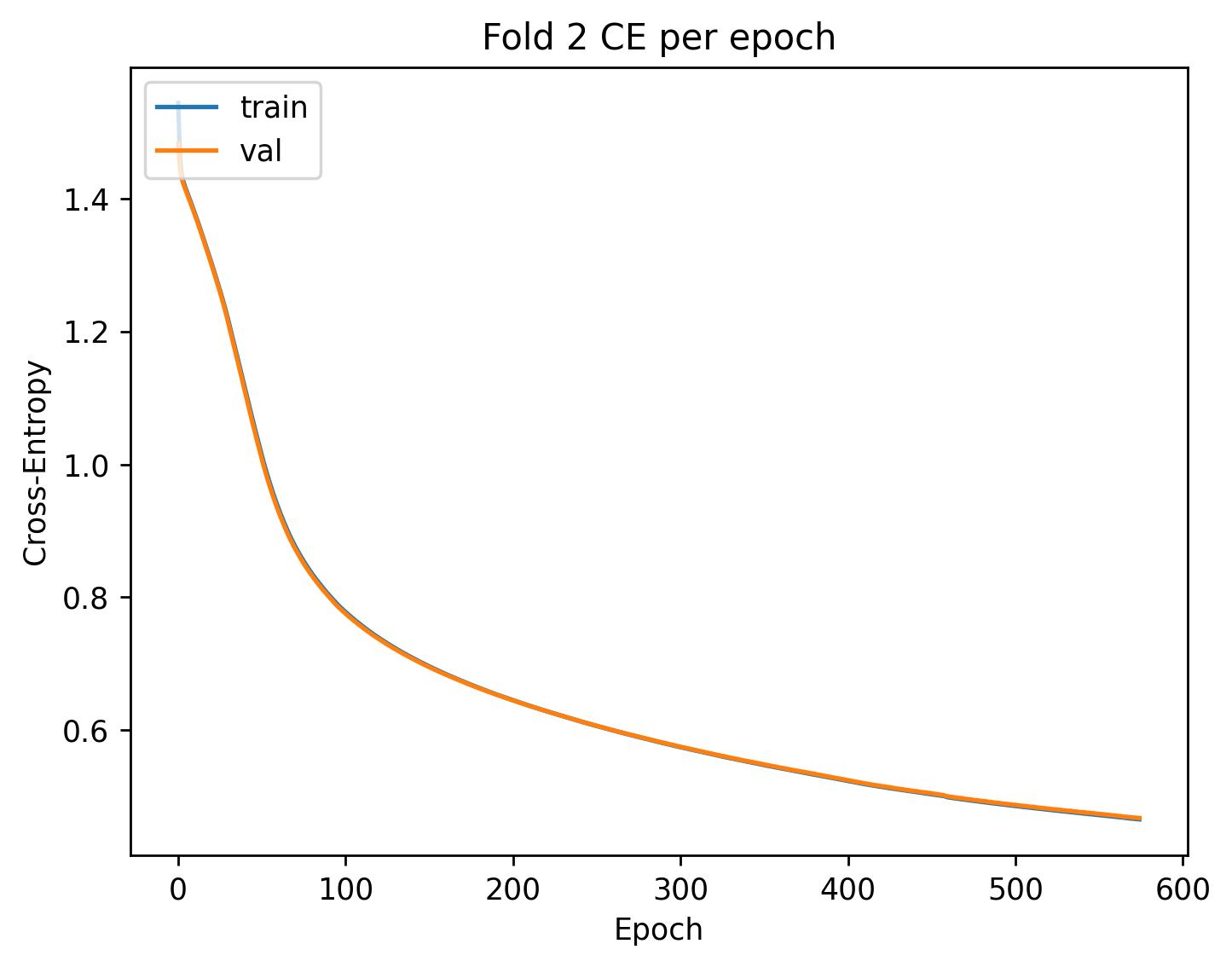
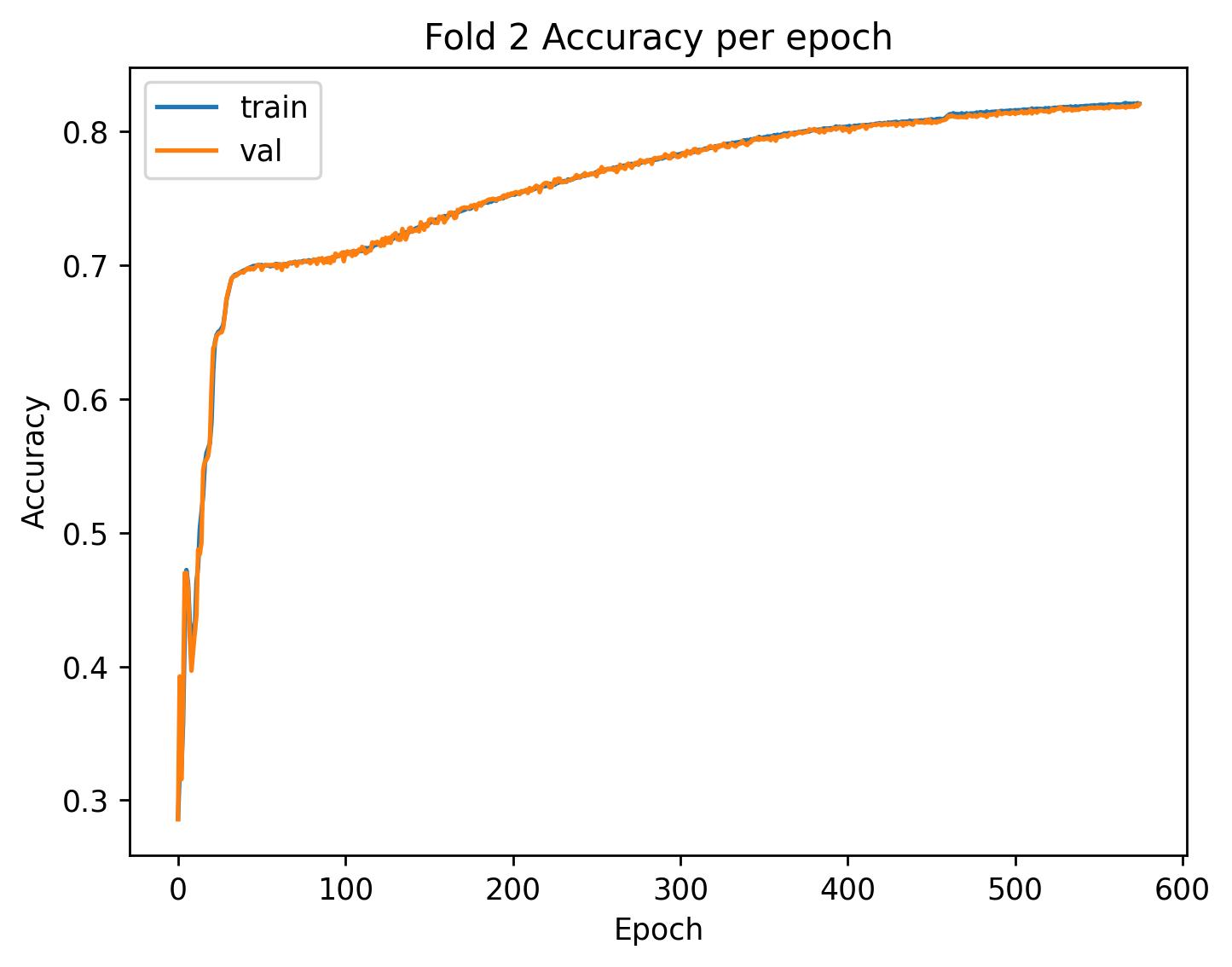
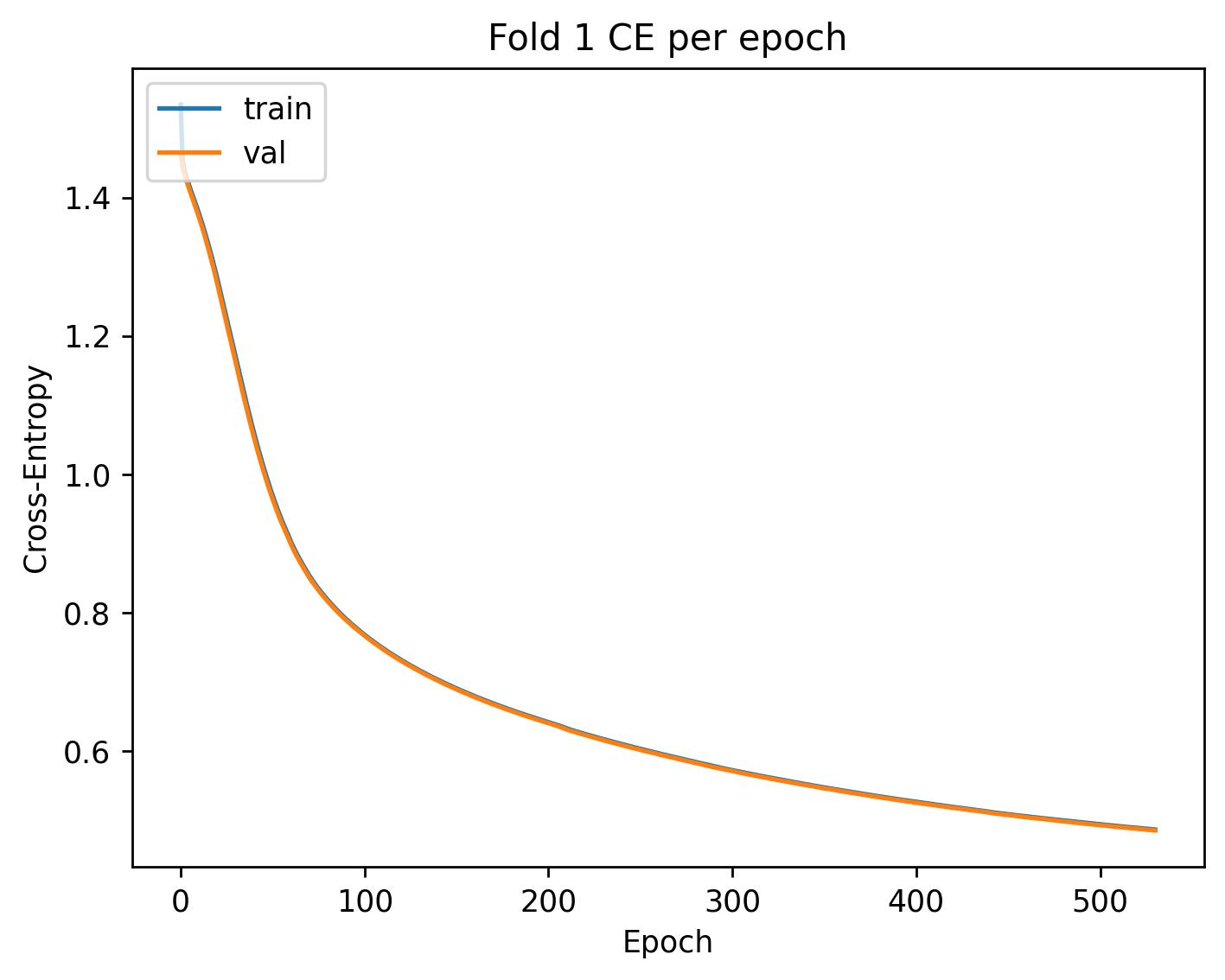
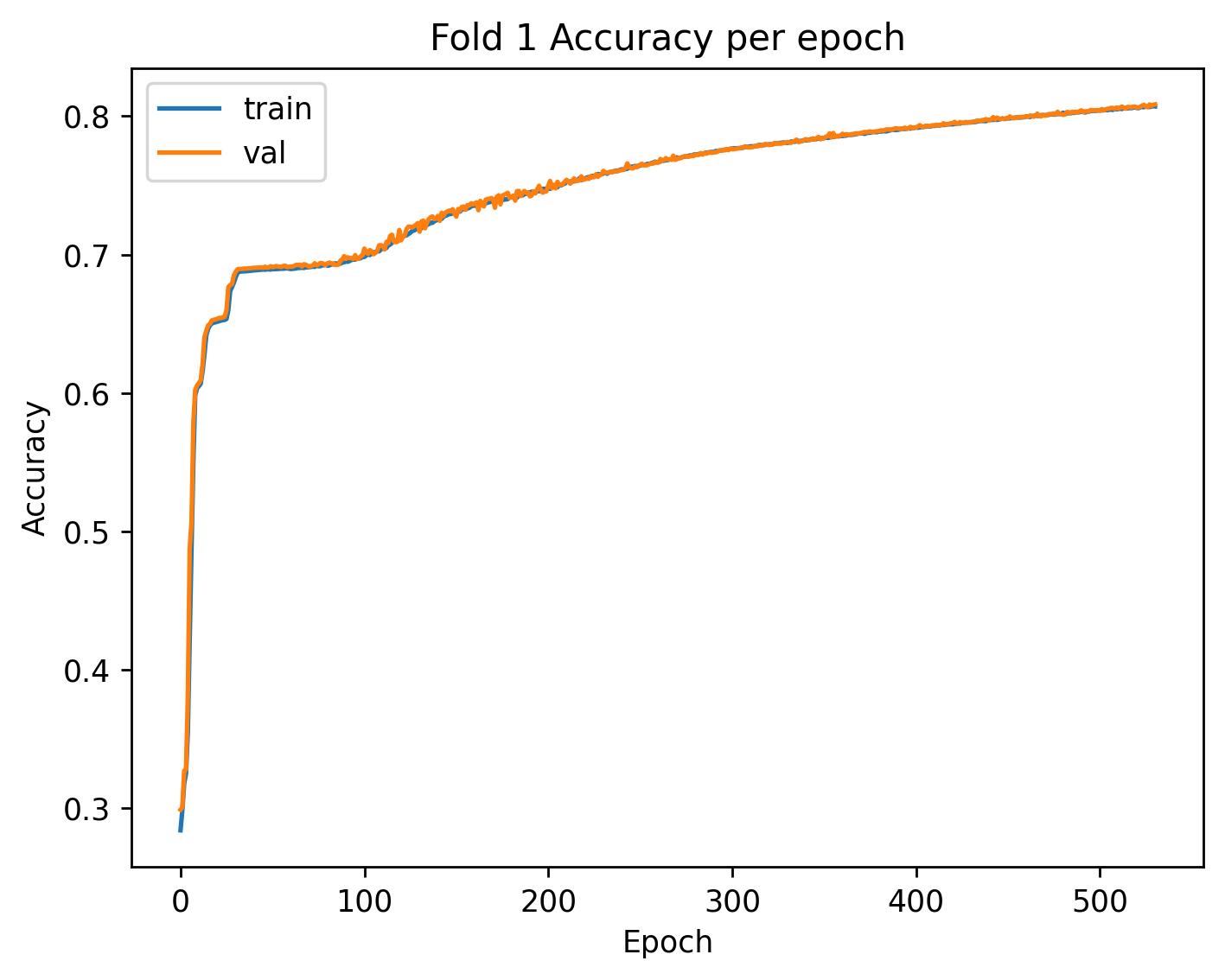
**5 νευρώνες στο κρυφό επίπεδο**



**12 νευρώνες στο κρυφό επίπεδο**



**23 νευρώνες στο κρυφό επίπεδο**



**Συμπεράσματα**

1. **Σχετικά με τον αριθμό των κρυφών κόμβων**

Οι κρυφοί κόμβοι βοηθούν το νευρωνικό δίκτυο να “ανακαλύψει” χαρακτηριστικά μεταξύ των δεδομένων της εισόδου που το βοηθούν να ταξινομήσει καλύτερα τα δεδομένα στην έξοδο. Ωστόσο η προσθήκη υπερβολικά πολλών κρυφών κόμβων μπορεί να οδηγήσει (εκτός από την αύξηση της πολυπλοκότητας του μοντέλου) σε μείωση της ικανότητας γενίκευσης, πράγμα μη επιθυμητό.

Από τα πειράματά μας παρατηρούμε ότι η αύξηση στους 12 από τους 5 νευρώνες ευνοεί πράγματι το δίκτυο. Φαίνεται πως η προσέγγιση μιας ακρίβειας γύρω στο 70% γίνεται ταχύτερα ενώ η μέση ακρίβεια αυξάνεται περίπου 4-5%. Παράλληλα η συνάρτηση κόστους φαίνεται να συγκλίνει σε μικρότερες τιμές με μεγαλύτερη ταχύτητα.

Αντιθέτως η αύξηση στους 23 νευρώνες δεν φαίνεται να προσφέρει τίποτα ιδιαίτερο. Δεν παρατηρείται αύξηση στην ταχύτητα σύγκλισης ούτε της ακρίβειας ταξινόμησης ούτε της συνάρτησης κόστους ενώ και οι μέσες τιμές παρουσιάζουν πολύ μικρές βελτιώσεις (στα όρια του λάθους). Συνεπώς το μόνο που προσφέρει αυτή η αύξηση είναι αύξηση πολυπλοκότητα και ενδεχομένως απώλεια ικανότητας γενίκευσης, κάτι το οποίο δεν παρατηρείται στα πειράματα ωστόσο.