Algoritem 1 *FederatedAverage.*

Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k

$\boldsymbol{FedAvg}()$:

Inicializacija globalnih uteži $w^{(1)}$

for $round\ r = 1, 2, \ldots, R$ **do**

$m = max(\boldsymbol{\lambda} \cdot N, 1)$

$S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$

for $client\ k \in S^{(r)}$ **in parallel do**

$\Delta \mathrm{w}_k^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)})$

$w^{(r+1)} = w^{(r)} + \eta_s \left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)}\, p_k\right]$, kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$

$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w})$:

$w_k = w$

$\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$

for $epoch\ e = 1, 2, \ldots, E_k$ **do**

for $batch\ b \in \mathcal{B}$ **do**

$w_k = w_k - \eta_k \nabla \ell(w_k;\ b)$

return $\Delta_k \leftarrow (w_k - w)$

Algoritem 2 *FedProx.*

Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k, $\boldsymbol{\mu}$ vpliv dodatnega člena

$\boldsymbol{FedProx}()$:

Inicializacija uteži $w^{(1)}$

for $round\ r = 1, 2, \ldots, R$ **do**

$m = max(\boldsymbol{\lambda} \cdot N, 1)$

$S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$

for $client\ k \in S^{(r)}$ **in parallel do**

$\Delta \mathrm{w}_k^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)})$

$w^{(r+1)} = w^{(r)} + \eta_s \left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)}\, p_k\right]$, kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$

$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w})$:

$w_k = w$

$\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$

for $epoch\ e = 1, 2, \ldots, E_k$ **do**

for $batch\ b \in \mathcal{B}$ **do**

$w_k = w_k - \eta_k \nabla\left(\ell(w_k;\ b) + \frac{\mu_k}{2} \|w_k - w\|^2\right)$

return $\Delta_k \leftarrow (w_k - w)$

Algoritem 4 SCAFFOLD: Stohastic Controlled Averaging for federated learning.
Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k

$\boldsymbol{SCAFFOLD}()$:
    Inicializacija $w^{(1)}, c^{(1)}$
    for $round\ r = 1, 2, \ldots, R$ **do**
        $m = max(\boldsymbol{\lambda} \cdot N, 1)$
        $S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$
        for $client\ k \in S^{(r)}$ **in parallel do**
            $\Delta \mathrm{w}_k^{(r)}, \Delta c_k^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)}, c^{(r)})$
        $w^{(r+1)} = w^{(r)} + \eta_s \left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} p_k\right]$, kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$
        $c^{(r+1)} = c^{(r)} + \frac{m}{N}\left[\sum_{k \in S^{(r)}} \Delta \mathrm{c}_k^{(r)} p_k\right]$, kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$

$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w}, \boldsymbol{c})$:
    $w_k = w$
    $\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$
    for $epoch\ e = 1, 2, \ldots, E_k$ **do**
        for $batch\ b \in \mathcal{B}$ **do**
            $w_k = w_k - \eta_k(\nabla \ell(w_k; b) - c_k + c)$
    $c_k^{+} = c_k - c + \frac{1}{E_k |\mathcal{B}| \eta_k}(w - w_k)$
    $(\Delta w_k, \Delta c_k) \leftarrow (w_k - w, c_k^{+} - c_k)$
    $c_k = c_k^{+}$
    return $(\Delta w_k, \Delta c_k)$

$$w_k = w_k - \eta_k \nabla \ell(w_k; b)$$

$$w^{(r+1)} = w^{(r)} + \eta_s \left[\sum\nolimits_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} p_k\right]$$

$$c^{(r+1)} = c^{(r)} + \frac{m}{N}\left[\sum\nolimits_{k \in S^{(r)}} \Delta \mathrm{c}_k^{(r)} p_k\right]$$

$$p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{k \in S^{(r)}} |\mathcal{D}_k|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$$

Algoritem 5 FedVARP: Variance Due to Partial Client Participation in Federated Learning.
Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k

$\boldsymbol{FedVARP}()$:
    Inicializacija $w^{(1)}$
    Inicializacija $y_k^{(1)} = 0;\ \forall \mathrm{k} \in [N]$
    for $round\ r\ =\ 1, 2, \ldots, R$ **do**
        $m = max(\boldsymbol{\lambda} \cdot N, 1)$
        $S^{(r)}\ \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$
        for $client\ k \in S^{(r)}$ **in parallel do**
            $\Delta \mathrm{w}_k^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)})$
        $v^{(r)} = \frac{1}{m}\left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} - y_k^{(r)}\right] + \frac{1}{N}\left[\sum_{j=1}^{N} y_j^{(r)}\right]$
        $w^{(r+1)} =\ w^{(r)} +\ \eta_s v^{(r)}$
        for $client\ k \in [N]$ **do**
            $y_k^{(r+1)} = \begin{cases} \Delta \mathrm{w}_k^{(r)}, & if\ k \in S^{(r)} \\ y_k^{(r)}, & otherwise \end{cases}$
$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w})$:
    $w_k \leftarrow w$
    $\mathcal{B}\ \leftarrow\ (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$
    for $epoch\ e = 1, 2, \ldots, E_k$ **do**
        for $batch\ b\ \in\ \mathcal{B}$ **do**
            $w_k\ \leftarrow\ w_k\ -\ \eta_k \nabla \ell(w_k;\ b)$
    return $\Delta_k \leftarrow (w_k - w)$

$$v^{(r)} = \frac{1}{m}\left[\sum\nolimits_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} - y_k^{(r)}\right] + \frac{1}{N}\left[\sum\nolimits_{j=1}^{N} y_j^{(r)}\right]$$

$$w^{(r+1)} =\ w^{(r)} +\ \eta_s v^{(r)}$$

$$y_k^{(r+1)} = \begin{cases} \Delta \mathrm{w}_k^{(r)}, & if\ k \in S^{(r)} \\ y_k^{(r)}, & otherwise \end{cases}$$

Algoritem 6 ClusterFedVARP: Variance Due to Partial Client Participation in Federated Learning.
Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_g}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k, $\boldsymbol{K}$ število gruč, $\mathbb{C}$ množica gruč, $\boldsymbol{c_k} \in [K]$ indeks gruče klienta k

$\boldsymbol{ClusterFedVARP}()$:
    Inicializacija $w^{(1)}$
    Inicializacija $y_k^{(1)} = 0;\ \forall \mathrm{k} \in [K]$
    (vsakemu klientu dodeli gručo $c_j \in [K]; \forall \mathrm{j} \in [N]$)
    for round r = $1, 2, \ldots, R$ **do**
        $m = max(\boldsymbol{\lambda} \cdot N, 1)$
        $S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$
        for client $k \in S^{(r)}$ **in parallel do**
            $\Delta \mathrm{w}_k^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)})$
        $v^{(r)} = \frac{1}{m}\left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} - y_{c_k}^{(r)}\right] + \frac{1}{N}\left[\sum_{j=1}^{N} y_{c_j}^{(r)}\right]$
        $w^{(r+1)} = w^{(r)} + \eta_s v^{(r)}$
        for $cluster\ \boldsymbol{k} \in [K]$ **do**
            $y_k^{(r+1)} = \begin{cases} \frac{\sum_{i\,\in\,S^{(r)} \cap \mathbb{C}_k} \Delta \mathrm{w}_i^{(r)}}{\left|S^{(r)} \cap \mathbb{C}_k\right|}, & if\ \left|S^{(r)} \cap \mathbb{C}_k\right| \neq 0 \\ y_k^{(r)}, & otherwise \end{cases}$
$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w})$:
    $w_k \leftarrow w$
    $\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$
    for $epoch\ e = 1, 2, \ldots, E_k$ **do**
        for $batch\ b \in \mathcal{B}$ **do**
            $w_k \leftarrow w_k - \eta_k \nabla \ell(w_k;\ b)$
    return $\Delta_k \leftarrow (w_k - w)$

$$v^{(r)} = \frac{1}{m}\left[\sum\nolimits_{k \in S^{(r)}} \Delta \mathrm{w}_k^{(r)} - y_{c_k}^{(r)}\right] + \frac{1}{N}\left[\sum\nolimits_{j=1}^{N} y_{c_j}^{(r)}\right]$$

$$y_k^{(r+1)} = \begin{cases} \dfrac{\sum_{i\,\in\,S^{(r)} \cap \mathbb{C}_k} \Delta \mathrm{w}_i^{(r)}}{\left|S^{(r)} \cap \mathbb{C}_k\right|}, & if\ \left|S^{(r)} \cap \mathbb{C}_k\right| \neq 0 \\ y_k^{(r)}, & otherwise \end{cases}$$

Algoritem 7 *FedRolex.*

Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{\beta}$ velikost nevronske mreže, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k

$\boldsymbol{FedAvg}()$:

Inicializacija uteži $w^{(1)}$

for round r = $1, 2, \ldots, R$ **do**

$m = max(\boldsymbol{\lambda} \cdot N, 1)$

$S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$

for client $k \in S^{(r)}$ **in parallel do**

(izberi pod-model glede na rundo in velikost, indeksi $I(r, \beta_k) \rightarrow w^{(r)}_{I(r,\beta_k)} \subseteq w^{(r)}$)

$\Delta \mathrm{w}^{(r)}_{k,I(r,\beta_k)} \leftarrow \boldsymbol{LocalSGD}(k, w^{(r)}_{I(r,\beta_k)})$

$$w^{(r+1)} = w^{(r)} + \frac{1}{\sum_{k \in S^{(r)}} p_k} \left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}^{(r)}_{k,I(r,\beta_k)}\, p_k \right],$$

kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$

$\boldsymbol{LocalSGD}(\boldsymbol{k}, \boldsymbol{w})$:

$w_k \leftarrow w$

$\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$

for epoch $e = 1, 2, \ldots, E_k$ **do**

for batch $b \in \mathcal{B}$ **do**

$w_k \leftarrow w_k - \eta_k \nabla \ell(w_k;\ b)$

return $\Delta w_k \leftarrow (w_k - w)$

Za vsak sloj se izbere premikajoče okno dolžine β_k sloja ki se začne na indeksu trenutne runde, v primeru konvolucijske nevronske mreže namesto posameznih uteži izbiramo filtre. Klient uči samo izbrane uteži

$$I(r, \beta) = \begin{cases} \{j, j+1, \ldots, j + \lfloor \beta_k |\mathcal{D}_k| \rfloor - 1\}, & if\ j + \lfloor \beta_k |\mathcal{D}_k| \rfloor \le |\mathcal{D}_k| \\ \{j, j+1, \ldots, |\mathcal{D}_k| - 1\} \cup \{0, 1, \ldots, j + \lfloor \beta_k |\mathcal{D}_k| \rfloor - 1 - |\mathcal{D}_k|\}, & else \end{cases}$$

Algoritem 8 FedSCAVAR

Vhodi: $\boldsymbol{\mathcal{D}}$ množica učnih podatkov, $\boldsymbol{N}$ število klientov, $\boldsymbol{\lambda}$ delež klientov ki prispeva k učenju, $\boldsymbol{B}$ velikost lokalnega minibatcha, $\boldsymbol{E}$ število epoh lokalnega učenja, $\boldsymbol{R}$ število ponovitev federativnega učenja, $\boldsymbol{\eta_s}$ hitrost učenja globalnega modela, $\boldsymbol{\eta_k}$ hitrost učenja klienta k, $\boldsymbol{K}$ število gruč, $\boldsymbol{C}$ množica gruč, $\boldsymbol{c_k}$ indeks gruče klienta k $\in [K]$

$\boldsymbol{SCAFFOLD}()$:

Inicializacija $w^{(1)}, c^{(1)}$

Inicializacija $y_k^{(1)} = 0;\ \forall k \in [K]$

(vsakemu klientu nastavi gručo $c_j \in [K]; \forall j \in [N]$)

for $round\ r = 1, 2, \ldots, R$ **do**

$m = max(\boldsymbol{\lambda} \cdot N, 1)$

$S^{(r)} \leftarrow (naključno\ izberi\ m\ klientov) \subseteq [N]$

for $client\ k \in S^{(r)}$ **in parallel do**

(izberi uteži iz globalnega modela glede na indekse $I(r, \beta_k) \rightarrow w_{I(r,\beta_k)}^{(r)} \subseteq w^{(r)}$)

(izberi uteži iz globalne kontrole glede na indekse $I(r, \beta_k) \rightarrow c_{I(r,\beta_k)}^{(r)} \subseteq c^{(r)}$)

$\Delta \mathrm{w}_{k,I(r,\beta_k)}^{(r)}, \Delta c_{k,I(r,\beta_k)}^{(r)} \leftarrow \boldsymbol{LocalSGD}(k, w_{I(r,\beta_k)}^{(r)}, c_{I(r,\beta_k)}^{(r)})$

$v^{(r)} = \frac{1}{m}\left[\sum_{k \in S^{(r)}} \Delta \mathrm{w}_{k,I(r,\beta_k)}^{(r)} - y_{c_k,I(r,\beta_k)}^{(r)}\right] + \frac{1}{N}\left[\sum_{j=1}^{N} y_{c_j}^{(r)}\right]$

$w^{(r+1)} = w^{(r)} + \eta_s v^{(r)}$

$c^{(r+1)} = c^{(r)} + \frac{m}{N}\left[\sum_{k \in S^{(r)}} \Delta \mathrm{c}_{k,I(r,\beta_k)}^{(r)} p_k\right]$, kjer je $p_k = \begin{cases} \frac{|\mathcal{D}_k|}{\sum_{i \in S^{(r)}} |\mathcal{D}_i|}, & option\ 1 \\ \frac{1}{m}, & option\ 2 \end{cases}$

for $cluster\ \boldsymbol{k} \in [K]$ **do**

$$y_k^{(r+1)} = \begin{cases} \frac{\sum_{i \in S^{(r)} \cap C_k} \Delta \mathrm{w}_i^{(r)}}{|S^{(r)} \cap C_k|}, & if\ \left|S^{(r)} \cap C_k\right| \neq 0 \\ y_k^{(r)}, & otherwise \end{cases}$$

$\boldsymbol{LocalSGD(k, w, c)}$:

$w_k = w$

$\mathcal{B} \leftarrow (razdeli\ \mathcal{D}_k\ na\ podmnožice\ velikosti\ B_k)$

for $epoch\ e = 1, 2, \ldots, E_k$ **do**

for $batch\ b \in \mathcal{B}$ **do**

$$w_k = w_k - \eta_k(\nabla(\ell(w_k;\ b) + \frac{\mu_k}{2}\|w_k - w\|^2) - c_k + c)$$

$c_k^{+} = c_k - c + \frac{1}{E_k|\mathcal{B}|\eta_k}(w - w_k)$

$(\Delta w_k, \Delta c_k) \leftarrow (w_k - w, c_k^{+} - c_k)$

$c_k = c_k^{+}$

return $(\Delta w_k, \Delta c_k)$